Benutzer- und Referenzhandbuch

Copyright © 1998–2010. Altova GmbH. All rights reserved. Use of this software is governed by and subject to an Altova software license agreement. XMLSpy, MapForce, StyleVision, SemanticWorks, SchemaAgent, UModel, DatabaseSpy, DiffDog, Authentic, AltovaXML, MissionKit, and ALTOVA as well as their logos are trademarks and/or registered trademarks of Altova GmbH.

XML, XSL, XHTML, and W3C are trademarks (registered in numerous countries) of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions, MIT, INRIA, and Keio. UNICODE and the Unicode Logo are trademarks of Unicode Inc. This software contains 3rd party copyrighted software or material that is protected by copyright and subject to other terms and conditions as detailed on the Altova website at http://www.altova.com/legal_3rdparty.html

ALTOVA®

Altova Authentic 2010 Browser Edition Benutzer- und Referenzhandbuch

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Published: 2010

© 2010 Altova GmbH

Inhaltsverzeichnis

1 Altova Authentic Browser Edition 3

2 Info zu dieser Dokumentation 6

3 Übersicht 8

3.1 Vorteile von Authentic Browser 9
3.2 Funktionsweise und Einrichten des Netzwerks 10
3.3 Implementierungsschritte 11

4 Einrichten des Servers 14

4.1 Authentic Browser Datei-Download 15
4.2 Authentic Browser Versionen 16
4.3 Konfigurieren des Browser-Diensts 17

5 Anmerkungen zum Einrichten des SPS 22

6 HTML-Seite für das Authentic Plug-in 24

6.1 Lizenzierung für die Enterprise Edition 25
6.2 Internet Explorer 27
6.2.1 Das OBJECT-Element 28
6.2.2 Das SCRIPT-Element 32
6.2.3 IE Beispiel 1: Einfach 33
6.2.4 IE Beispiel 2: Tabelle sortieren 35
6.2.5 IE Beispiel 3: DB-Referenz 38
6.3 Firefox 39
6.3.1 Das EMBED-Element 40
6.3.2 Hinzufügen von Event Listener-Funktionen 44
6.3.3 Firefox Beispiel 1: Einfach 45
6.3.4 Firefox Beispiel 2: Tabelle sortieren 47
6.4 Browser-unabhängig 50

6.4.1 Abrufen des richtigen Plug-in ... 51
6.4.2 Browser-unabhängiges Beispiel ... 53

7 Benutzerreferenz 58

7.1 Mechanismen ... 59
7.1.1 Ereignisse: Connection Point (IE-spezifisch) ... 60
7.1.2 Ereignisse: Hinzufügen von Event Listener-Funktionen (Firefox-spezifisch 62
7.1.3 Ereignisse: Symbolleisten-Schaltfläche ... 63
7.1.4 Ereignisse: Referenz ... 64
7.1.5 Aufrufen und Ändern von Dokumentinhalt ... 65
7.1.6 Bearbeitungsoperationen ... 66
7.1.7 Suchen und Ersetzen ... 67
7.1.8 Zeilenoperationen ... 68
7.1.9 Tastaturkürzel ... 69
7.1.10 Textstatus-Schaltflächen ... 70
7.1.11 Eingabehilfen ... 71
7.1.12 Pakete ... 72
7.1.13 Verwendung von XMLData ... 74
7.1.14 DOM und XMLData ... 78
7.2 Objekte ... 81
7.2.1 Authentic ... 82
- Authentic.ApplyTextState ... 85
- Authentic.attachCallBack ... 86
- AuthenticView ... 88
- Authentic.AutoHideUnusedCommandGroups ... 89
- Authentic.BaseURL ... 90
- Authentic.ClearSelection ... 91
- Authentic.ClearUndoRedo ... 92
- Authentic.ControlInitialized ... 93
- Authentic.CreateChild ... 94
- Authentic.CurrentSelection ... 95
- Authentic.DesignDataLoadObject ... 96
- Authentic.EditClear ... 97
- Authentic.EditCopy ... 98
- Authentic.EditCut ... 99
- Authentic.EditPaste ... 100
- Authentic.EditRedo ... 101
- Authentic.EditSelectAll ... 102
- Authentic.EditUndo ... 103
- Authentic.EnableModifications ... 104
- Authentic.EntryHelperAlignment ... 105
- Authentic.EntryHelpersEnabled ... 106

– Authentic.EntryHelperSize ... 107
– Authentic.EntryHelperWindows ... 108
– Authentic.event ... 109
– Authentic.FindDialog ... 110
– Authentic.FindNext ... 111
– Authentic.GetAllAttributes ... 112
– Authentic.GetAllowedElements ... 114
– Authentic.GetFileVersion ... 116
– Authentic.GetNextVisible ... 117
– Authentic.GetPreviousVisible ... 118
– Authentic.IsEditClearEnabled ... 119
– Authentic.IsEditCopyEnabled ... 120
– Authentic.IsEditCutEnabled ... 121
– Authentic.IsEditPasteEnabled ... 122
– Authentic.IsEditRedoEnabled ... 123
– Authentic.IsEditUndoEnabled ... 124
– Authentic.IsFindNextEnabled ... 125
– Authentic.IsRowAppendEnabled ... 126
– Authentic.IsRowDeleteEnabled ... 127
– Authentic.IsRowDuplicateEnabled ... 128
– Authentic.IsRowInsertEnabled ... 129
– Authentic.IsRowMoveDownEnabled ... 130
– Authentic.IsRowMoveUpEnabled ... 131
– Authentic.IsTextStateApplied ... 132
– Authentic.IsTextStateEnabled ... 133
– Authentic.LoadXML ... 134
– Authentic.MarkUpView ... 135
– Authentic.Modified ... 136
– Authentic.Print ... 137
– Authentic.PrintPreview ... 138
– Authentic.RedrawEntryHelpers ... 139
– Authentic.ReloadToolbars ... 140
– Authentic.ReplaceDialog ... 141
– Authentic.Reset ... 142
– Authentic.RowAppend ... 143
– Authentic.RowDelete ... 144
– Authentic.RowDuplicate ... 145
– Authentic.RowInsert ... 146
– Authentic.RowMoveDown ... 147
– Authentic.RowMoveUp ... 148
– Authentic.Save ... 149
– Authentic.SaveButtonAutoEnable ... 150
– Authentic.SavePOST ... 151
– Authentic.SaveXML ... 152
– Authentic.SchemaLoadObject ... 153

– Authentic.SelectionChanged ... 154
– Authentic.SelectionMoveTabOrder ... 155
– Authentic.SelectionSet ... 156
– Authentic.SetUnmodified ... 157
– Authentic.StartEditing ... 158
– Authentic.StartSpellChecking ... 159
– Authentic.TextStateBmpURL ... 160
– Authentic.TextStateToolbarLine ... 161
– Authentic.ToolbarRows ... 162
– Authentic.ToolbarsEnabled ... 163
– Authentic.ToolbarTooltipsEnabled ... 164
– Authentic.UICommands ... 165
– Authentic.ValidateDocument ... 166
– Authentic.validationBadData ... 167
– Authentic.validationMessage ... 168
– Authentic.XMLDataLoadObject ... 169
– Authentic.XMLDataSaveUrl ... 170
– Authentic.XMLRoot ... 171
– Authentic.XMLTable ... 172

7.2.2 AuthenticCommand ... 173
– AuthenticCommand.CommandID ... 174
– AuthenticCommand.Group ... 175
– AuthenticCommand.ShortDescription ... 176
– AuthenticCommand.Name ... 177

7.2.3 AuthenticCommands ... 178
– AuthenticCommands.Count ... 179
– AuthenticCommands.Item ... 180

7.2.4 AuthenticDataTransfer ... 181
– AuthenticDataTransfer.dropEffect ... 182
– AuthenticDataTransfer.getData ... 183
– AuthenticDataTransfer.ownDrag ... 184
– AuthenticDataTransfer.type ... 185

7.2.5 AuthenticEvent ... 186
– AuthenticEvent.altKey ... 187
– AuthenticEvent.altLeft ... 188
– AuthenticEvent.button ... 189
– AuthenticEvent.cancelBubble ... 190
– AuthenticEvent.clientX ... 191
– AuthenticEvent.clientY ... 192
– AuthenticEvent.ctrlKey ... 193
– AuthenticEvent.ctrlLeft ... 194
– AuthenticEvent.dataTransfer ... 195
– AuthenticEvent.fromElement ... 196
– AuthenticEvent.keyCode ... 197
– AuthenticEvent.propertyName ... 198

– AuthenticEvent.repeat ... 199
– AuthenticEvent.returnValue ... 200
– AuthenticEvent.shiftKey ... 201
– AuthenticEvent.shiftLeft ... 202
– AuthenticEvent.srcElement ... 203
– AuthenticEvent.type ... 204
7.2.6 AuthenticLoadObject ... 205
– AuthenticLoadObject.String ... 206
– AuthenticLoadObject.URL ... 207
7.2.7 AuthenticRange ... 208
– AuthenticRange.AppendRow ... 210
– AuthenticRange.Application ... 211
– AuthenticRange.CanPerformAction ... 212
– AuthenticRange.CanPerformActionWith ... 213
– AuthenticRange.Clone ... 214
– AuthenticRange.CollapsToBegin ... 215
– AuthenticRange.CollapsToEnd ... 216
– AuthenticRange.Copy ... 217
– AuthenticRange.Cut ... 218
– AuthenticRange.Delete ... 219
– AuthenticRange.DeleteRow ... 220
– AuthenticRange.DuplicateRow ... 221
– AuthenticRange.ExpandTo ... 222
– AuthenticRange.FirstTextPosition ... 223
– AuthenticRange.FirstXMLData ... 224
– AuthenticRange.FirstXMLDataOffset ... 225
– AuthenticRange.GetElementAttributeNames ... 227
– AuthenticRange.GetElementAttributeValue ... 228
– AuthenticRange.GetElementHierarchy ... 229
– AuthenticRange.GetEntityNames ... 230
– AuthenticRange.Goto ... 231
– AuthenticRange.GotoNext ... 232
– AuthenticRange.GotoNextCursorPosition ... 233
– AuthenticRange.GotoPrevious ... 234
– AuthenticRange.GotoPreviousCursorPosition ... 235
– AuthenticRange.HasElementAttribute ... 236
– AuthenticRange.InsertEntity ... 237
– AuthenticRange.InsertRow ... 238
– AuthenticRange.IsEmpty ... 239
– AuthenticRange.IsEqual ... 240
– AuthenticRange.IsInDynamicTable ... 241
– AuthenticRange.LastTextPosition ... 242
– AuthenticRange.LastXMLData ... 243
– AuthenticRange.LastXMLDataOffset ... 244
– AuthenticRange.MoveBegin ... 246

– AuthenticRange.MoveEnd ... 247
– AuthenticRange.MoveRowDown ... 248
– AuthenticRange.MoveRowUp ... 249
– AuthenticRange.Parent ... 250
– AuthenticRange.Paste ... 251
– AuthenticRange.PerformAction ... 252
– AuthenticRange.Select ... 253
– AuthenticRange.SelectNext ... 254
– AuthenticRange.SelectPrevious ... 255
– AuthenticRange.SetElementAttributeValue ... 256
– AuthenticRange.SetFromRange ... 257
– AuthenticRange.Text ... 258
7.2.8 AuthenticSelection ... 259
– AuthenticSelection.End ... 260
– AuthenticSelection.EndTextPosition ... 261
– AuthenticSelection.Start ... 262
– AuthenticSelection.StartTextPosition ... 263
7.2.9 AuthenticToolbarButton ... 264
– AuthenticToolbarButton.CommandID ... 265
7.2.10 AuthenticToolbarButtons ... 266
– AuthenticToolbarButtons.Count ... 267
– AuthenticToolbarButtons.Item ... 268
– AuthenticToolbarButtons.NewButton ... 269
– AuthenticToolbarButtons.NewCustomButton ... 270
– AuthenticToolbarButtons.NewSeparator ... 271
– AuthenticToolbarButtons.Remove ... 272
7.2.11 AuthenticToolbarRow ... 273
– AuthenticToolbarRowAlignment ... 274
– AuthenticToolbarRowButtons ... 275
7.2.12 AuthenticToolbarRows ... 276
– AuthenticToolbarRows.Count ... 277
– AuthenticToolbarRows.Item ... 278
– AuthenticToolbarRows.RemoveRow ... 279
– AuthenticToolbarRows.NewRow ... 280
7.2.13 AuthenticView ... 281
– AuthenticView.Application ... 282
– AuthenticView.DocumentBegin ... 283
– AuthenticView.DocumentEnd ... 284
– AuthenticView.Goto ... 285
– AuthenticView.MarkupVisibility ... 286
– AuthenticView.Parent ... 287
– AuthenticView.Print ... 288
– AuthenticView.Redo ... 289
– AuthenticView.Selection ... 290

– AuthenticView.Undo ... 291
– AuthenticView.UpdateXMLInstanceEntities ... 292
– AuthenticView.WholeDocument ... 293
7.2.14 AuthenticXMLTableCommands ... 294
– AuthenticXMLTableCommands.AlignHorizontalCenter ... 296
– AuthenticXMLTableCommands.AlignHorizontalJustify ... 297
– AuthenticXMLTableCommands.AlignHorizontalLeft ... 298
– AuthenticXMLTableCommands.AlignHorizontalRight ... 299
– AuthenticXMLTableCommands.AlignVerticalBottom ... 300
– AuthenticXMLTableCommands.AlignVerticalCenter ... 301
– AuthenticXMLTableCommands.AlignVerticalTop ... 302
– AuthenticXMLTableCommands.AppendCol ... 303
– AuthenticXMLTableCommands.AppendRow ... 304
– AuthenticXMLTableCommands.Delete ... 305
– AuthenticXMLTableCommands.DeleteCol ... 306
– AuthenticXMLTableCommands.DeleteRow ... 307
– AuthenticXMLTableCommands.EditProperties ... 308
– AuthenticXMLTableCommands.Insert ... 309
– AuthenticXMLTableCommands.InsertCol ... 310
– AuthenticXMLTableCommands.InsertRow ... 311
– AuthenticXMLTableCommands.JoinDown ... 312
– AuthenticXMLTableCommands.JoinLeft ... 313
– AuthenticXMLTableCommands.JoinRight ... 314
– AuthenticXMLTableCommands.JoinUp ... 315
– AuthenticXMLTableCommands.MayAlignHorizontal ... 316
– AuthenticXMLTableCommands.MayAlignVertical ... 317
– AuthenticXMLTableCommands.MayAppendCol ... 318
– AuthenticXMLTableCommands.MayAppendRow ... 319
– AuthenticXMLTableCommands.MayDelete ... 320
– AuthenticXMLTableCommands.MayDeleteCol ... 321
– AuthenticXMLTableCommands.MayDeleteRow ... 322
– AuthenticXMLTableCommands.MayEditProperties ... 323
– AuthenticXMLTableCommands.MayInsert ... 324
– AuthenticXMLTableCommands.MayInsertCol ... 325
– AuthenticXMLTableCommands.MayInsertRow ... 326
– AuthenticXMLTableCommands.MayJoinDown ... 327
– AuthenticXMLTableCommands.MayJoinLeft ... 328
– AuthenticXMLTableCommands.MayJoinRight ... 329
– AuthenticXMLTableCommands.MayJoinUp ... 330
– AuthenticXMLTableCommands.MaySplitHorizontal ... 331
– AuthenticXMLTableCommands.MaySplitVertical ... 332
– AuthenticXMLTableCommands.SplitHorizontal ... 333
– AuthenticXMLTableCommands.SplitVertical ... 334
7.2.15 XMLData ... 335
– XMLData.AppendChild ... 336

- XMLData.CountChildren ... 337
- XMLData.CountChildrenKind ... 338
- XMLData.EraseAllChildren ... 339
- XMLData.EraseCurrentChild ... 340
- XMLData.GetChild ... 341
- XMLData.GetChildKind ... 342
- XMLData.GetCurrentChild ... 343
- XMLData.GetFirstChild ... 344
- XMLData.GetNextChild ... 345
- XMLData.HasChildren ... 346
- XMLData.HasChildrenKind ... 347
- XMLData.InsertChild ... 348
- XMLData.IsSameNode ... 349
- XMLData.Kind ... 350
- XMLData.MayHaveChildren ... 351
- XMLData.Name ... 352
- XMLData.Parent ... 353
- XMLData.TextValue ... 354

7.3 Enumerationen ... 355
7.3.1 SPYXMLDataKind ... 356
7.3.2 SPYAuthenticElementActions ... 357
7.3.3 SPYAuthenticCommand ... 358
7.3.4 SPYAuthenticCommandGroup ... 360
7.3.5 SPYAuthenticToolbarAllignment ... 361
7.3.6 SPYAuthenticEntryHelperWindows ... 362
7.3.7 SPYAuthenticElementKind ... 363
7.3.8 SPYAuthenticActions ... 364
7.3.9 SPYAuthenticDocumentPosition ... 365
7.3.10 SPYAuthenticMarkupVisibility ... 366

8 Lizenzinformationen 368

8.1 Electronic Software Distribution ... 369
8.2 Rechte am geistigen Eigentum ... 370
8.3 Endbenutzer-Lizenzvereinbarung ... 371

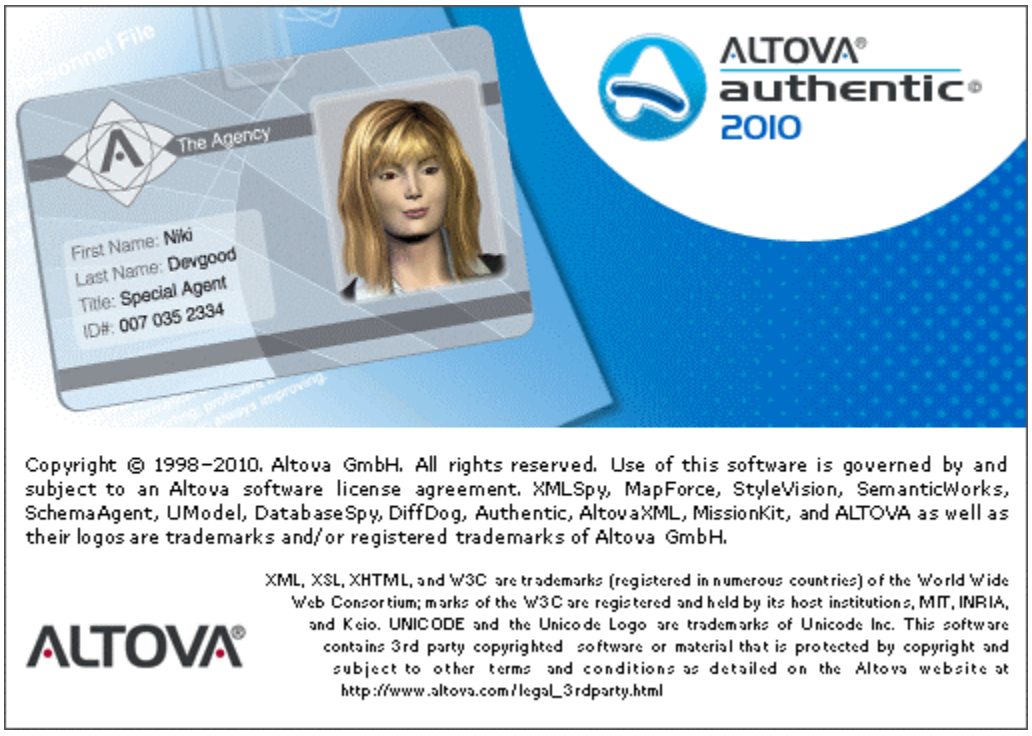
Index

Kapitel 1

Altova Authentic Browser Edition

Altova Authentic Browser Edition

Mit der **Altova® Authentic® 2010 Browser Edition** können Benutzer XML- und Datenbank-Content über eine Benutzeroberfläche bearbeiten, die einen Textverarbeitungsprogramm sehr ähnlich sieht. Die Browser Edition kann in jede beliebige Webseite eingebettet werden und ermöglicht die Bearbeitung direkt über den Browser. Die Authentic Browser Edition steht als Plug-in für **Internet Explorer** und **Firefox** zur Verfügung.


The Agency
First Name: Niki
Last Name: Devgood
Title: Special Agent
ID#: 007 035 2334
ALTOVA®
authentic®
2010
Copyright © 1998–2010. Altova GmbH. All rights reserved. Use of this software is governed by and subject to an Altova software license agreement. XMLSpy, MapForce, StyleVision, SemanticWorks, SchemaAgent, UModel, DatabaseSpy, DiffDog, Authentic, AltovaXML, MissionKit, and ALTOVA as well as their logos are trademarks and/or registered trademarks of Altova GmbH.
ALTOVA®
XML, XSL, XHTML, and W3C are trademarks (registered in numerous countries) of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions, MIT, INRIA, and Keio. UNICODE and the Unicode Logo are trademarks of Unicode Inc. This software contains 3rd party copyrighted software or material that is protected by copyright and subject to other terms and conditions as detailed on the Altova website at http://www.altova.com/legal_3rdparty.html

Es gibt nun zwei Versionen von Authentic Browser: (i) eine **Enterprise Edition**, die komplexe Features unterstützt und für die eine Lizenz erworben werden muss; und (ii) eine **Community Edition**, die kostenlos zur Verfügung steht.

Kapitel 2

Info zu dieser Dokumentation

Info zu dieser Dokumentation

Diese Dokumentation ist das Handbuch zu Authentic Browser und in die folgenden Abschnitte gegliedert:

- eine Übersicht mit Erläuterungen zu: (i) den Vorteilen der Verwendung von Authentic Browser; (ii) der Funktionsweise von Authentic Browser und dem Einrichten eines Authentic Browser Netzwerks; sowie (iii) den Implementiertungsschritten für ein Authentic Browser Projekt.
- einen Abschnitt Einrichten des Servers, in dem beschrieben wird, wie Sie einen Server für ein Authentic Browser Projekt einrichten.
- eine ausführliche Beschreibung der HTML-Seite für das Authentic Plug-in. In diesem Abschnitt wird erläutert, wie die HTML-Seite erstellt wird, die der Client öffnen muss, um auf das XML-Dokument zugreifen zu können. Da jeder unterstützte Browser ein anderes Verfahren verwendet, werden die einzelnen HTML-Seiten für jeden Browser separat beschrieben: Internet Explorer und Firefox.
- einen aus drei Teilen bestehenden Referenzabschnitt (Benutzerreferenz: Mechanismen , Benutzerreferenz: Objekte und Benutzerreferenz: Enumerationen). Hier werden die Mechanismen, Objekte und Enumerationen beschrieben, mit Hilfe derer Sie die Authentic-Ansicht in Authentic Browser erstellen können.

Weitere Dokumentation zu diesem Thema
Zu diesem Thema gibt es zwei weitere Handbücher von Altova:

- Informationen zum Erstellen von StyleVision Power Stylesheets (SPSs) finden Sie auf der Altova Website in der Dokumentation zum Altova Produkt StyleVision. Ein SPS ist die Datei, in der definiert ist, wie die Authentic-Ansicht eines XML-Dokuments aussieht. Diese Dokumentation ist relevant, wenn Sie die Authentic-Ansichtsseite für die XML-Bearbeitung erstellen.
- Informationen zur Verwendung der Authentic-Ansicht, die Sie im Benutzerhandbuch im Tutorial zur Authentic -Ansicht und den Abschnitten über die Verwendung von Authentic Desktop finden. Das Benutzerhandbuch finden Sie auf der Altova Website.

© 2010 Altova GmbH

Kapitel 3

Übersicht

Übersicht

Mit der **Altova® Authentic® 2010 Browser Edition** können Benutzer XML-Dokumente bearbeiten, die auf **StyleVision Power Stylesheets (.`sps`-Dateien) basieren, die mit StyleVision erstellt wurden.** Authentic Browser ist eine einzigartige Lösung, die es Ihnen gestattet, XML-Inhalte auf jedem beliebigen Desktop (Client) in Ihrem Unternehmen in einem Internet Browser-Fenster zu bearbeiten.

Diese Übersicht ist in die folgenden Unterabschnitte gegliedert:

- Vorteile von Authentic Browser: Hier finden Sie eine Auflistung der Verwendungsmöglichkeiten und Vorteile von Authentic Browser.
- Funktionsweise und Einrichten im Netzwerk: Dieser Abschnitt enthält eine kurze Beschreibung der Funktionsweise der Authentic Browser Lösung und der Netzwerk-Architektur, die für die Implementierung der Lösung erforderlich ist.
- Implementierungsschritte: Listet die einzelnen Schritte auf, die erforderlich sind, um ein Authentic Browser Projekt zu implementieren.

© 2010 Altova GmbH

1 Vorteile von Authentic Browser

Die Authentic Browser Lösung zur Bearbeitung von XML-Dateien weist eine Reihe von Vorteilen auf. Einige davon sind unten aufgelistet:

- Ermöglicht mehreren Benutzern den Zugriff und die Bearbeitung eines XML-Dokuments über ihre Browser. Derzeit kann Authentic Browser mit **Internet Explorer 5.5 oder höher** und **Firefox 3.0 oder höher** verwendet werden.
- Vereinfacht das firmenweite Deployment und die Wartung von Applikationen enorm und senkt gleichzeitig die Gesamtinvestitionskosten.
- Basiert auf offenen Standards wie XML Schema und XSLT.
- Ist vollständig Unicode-kompatibel.
- Verwendet die Authentic-Ansicht, die auf dem gebräuchlichen Internet Explorer Browser oder Firefox basiert (IE bzw. Firefox muss auf einem Client-Rechner installiert sein, damit die Authentic-Ansicht funktioniert). In der Authentic-Ansicht können XML-Dateien auf WYSIWYG Art bearbeitet werden, d.h. der Benutzer muss den zugrunde liegenden XML-Code nicht anzeigen lassen.
- Benötigt keine weitere Software, da das Authentic Browser Plug-in in Internet Explorer oder Firefox integriert wird.
- Authentic Browser ist ein ActiveX Control, wobei die COM-Schnittstelle über das Authentic Objekt definiert wird. Das vollständige Objektmodell ist im Abschnitt Benutzerreferenz: Objekte dieser Dokumentation beschrieben.

2 Funktionsweise und Einrichten des Netzwerks

In diesem Abschnitt wird beschrieben, wie Authentic Browser funktioniert und wie man das Netzwerk für ein Authentic Browser Projekt einrichtet.

Funktionsweise
Authentic Browser funktioniert im Prinzip folgendermaßen:

- Das Installationspaket des Authentic Browser Plug-in wird auf einem Server in Ihrem Unternehmen gespeichert. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss das Paket auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde.
- Es wird automatisch auf einem Client-Rechner installiert, wenn der Client eine HTML-Seite öffnet, die das Plug-in aufruft. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss die HTML-Seite auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde.
- Sobald das Plug-in auf dem Client installiert ist, hat der Code in der HTML-Seite zur Folge, dass im Browser-Fenster ein Authentic-Ansichtsbearbeitungsfenster geöffnet wird.
- Das zu ladende XML-Dokument wird im Fenster der Authentic-Ansicht geöffnet und der Benutzer kann mit der Bearbeitung beginnen und die Änderungen direkt im XML-Dokument speichern.

Einrichten des Netzwerks
In den folgenden Punkten wird beschrieben, wie das Netzwerk konfiguriert werden muss, damit ein Authentic Browser-Projekt erfolgreich implementiert werden kann:

- Die Authentic Browser Edition steht nur für die Windows-Plattform zur Verfügung und funktioniert auf Windows XP, Windows Vista und Windows 7. Es gibt nun zwei Versionen von Authentic Browser: (i) eine **Enterprise Edition**, die komplexe Features unterstützt und für die eine Lizenz erworben werden muss; und (ii) eine **Community Edition**, die kostenlos zur Verfügung steht.
- Ein Rechner wird als Server benötigt. Die Authentic Browser Installationsdatei wird auf dem Server gespeichert.
- Normalerweise werden auch die XML-Datei/en, die SPS-Datei und die XML-Schema-Datei/en auf dem Server gespeichert, können aber auch unter einem anderen Pfad gespeichert werden.
- Der Server wird mit beliebig vielen Client-Rechnern verbunden.
- Jeder Client-Rechner muss einen der folgenden Browser installiert haben: **Internet Explorer 5.5 oder höher** oder **Firefox 3.0 oder höher**. Der Browser muss die HTML-Seite für Authentic öffnen und die in dieser Seite kodierten Funktionen ausführen. Das Authentic Browser Plug-in steht sowohl für die 32-Bit- als auch für die 64-Bit Internet Explorer Version zur Verfügung.
- Auf jedem Client-Rechner muss **Internet Explorer 5.5 oder höher** installiert sein. Dies liegt daran, dass die Benutzeroberfläche der Authentic-Ansicht (die im Browser-Fenster angezeigt wird) mit Internet Explorer generiert wird.
- Um die Authentic Browser Enterprise Edition verwenden zu können, benötigen Sie einen Lizenzschlüssel. Nähere Informationen dazu finden Sie unter Lizenzierung für die Enterprise Edition.

© 2010 Altova GmbH

3 Implementierungsschritte

Um ein Authentic Browser-Projekt implementieren zu können, müssen Sie folgende Schritte durchführen:

1. Downloaden Sie die gezippte Authentic Browser Datei (CAB und/oder XPI) von der Altova Website und speichern Sie diese in einem Ordner auf Ihrem Server. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss das Paket auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde. Dieser Teil des Vorgangs ist im Abschnitt Einrichten des Servers beschrieben. Authentic Browser wird aus dieser gezippten Datei auf den verschiedenen Client-Rechnern im Netzwerk installiert. Wenn ein Client diese gezippte Datei (über das HTML-Seite für Authentic Plug-in) zum ersten Mal aufruft, wird Authentic Browser automatisch auf den Client-Rechner heruntergeladen, entzippt und auf dem Client installiert.
2. Erstellen Sie die HTML-Datei, die das Authentic Objekt enthält. Wir nennen diese Datei die HTML-Seite für das Authentic Plug-in. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss die HTML-Seite auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde. Auf einem Client wird diese Seite in einem Browser (Internet Explorer oder Firefox) geöffnet. Die HTML-Seite enthält nicht nur Code, der Authentic Browser (wie in Schritt 1 oben beschrieben) installiert. Sie öffnet im Browser-Fenster auch ein XML-Dokument in einer Authentic -Ansicht. Die HTML-Seite muss daher korrekt geschrieben sein, damit diese Funktionen ausgeführt werden können. Eine Anleitung zur Erstellung der HTML-Seite finden Sie im Abschnitt HTML-Seite für das Authentic Plug-in.
3. Erstellen Sie das XML-Schema, das StyleVision Power Stylesheet (SPS) und die XML-Datei (oder XML-Vorlagendatei). Die mit Altova StyleVision erstellte SPS-Datei definiert, wie die XML-Datei in der Authentic-Ansicht angezeigt wird und wie Daten in die XML-Datei eingegeben werden. Die XML-Datei ist die Datei, die mit Authentic Browser bearbeitet wird. Nähere Informationen zum Erstellen der SPS-Datei und zum Verknüpfen einer XML-Datei damit finden Sie in der Benutzerdokumentation zu Altova StyleVision.
4. Für einige für das Authentic Browser Plug-in erstellte SPS-Dateien werden unter Umständen weitere Informationen oder Einstellungen benötigt. So müssen z.B. bei Verwendung von datenbankbasierten SPS-Dateien bestimmte Einstellungen im SPS, auf dem Server und auf den Clients vorgenommen werden. Diese zusätzlichen Einstellungen sind im Abschnitt Anmerkungen zum SPS näher beschrieben.
5. Nach Einrichten des Servers und Erstellung der HTML-Seite, sollte die gesamte Installation getestet werden. Verwenden Sie zu Testzwecken eines der Beispiele aud dem Abschnitt HTML -Seite für das Authentic Plug-in. Denken Sie daran, die erforderliche XSD-, XML- und SPS-Datei in den richtigen Ordner zu kopieren, der in der HTML-Datei referenziert ist.

Anmerkung: Wenn der Benutzer eine HTML-Datei, die ein Authentic-Objekt enthält, zum ersten Mal öffnet, wird das Authentic Browser Plug-in vom Server heruntergeladen und als Plug-in zur Browser-Applikation des Client installiert. Es sind keine separaten oder zusätzlichen Installationschritte erforderlich.

Kapitel 4

Einrichten des Servers

Einrichten des Servers

Um Authentic Browser auf Ihrem Server zu installieren, gehen Sie folgendermaßen vor:

1. Installieren und konfigurieren Sie den Browser-Dienst Ihres Servers. Wenn Sie Internet Information Services (IIS) von Microsoft verwenden, beachten Sie, dass bei der Installation ein Standardverzeichnis mit dem Namen `Inetpub` sowie Unterverzeichnisse angelegt werden. Das Root-Verzeichnis des Servers wäre dann: `//Inetpub/wwwroot`. Dies ist das Verzeichnis, das Sie mit der IP-Adresse des Servers erreichen, wenn Sie (in Ihrer Browser Service-Konfiguration) kein anderes als das Root-Verzeichnis angeben. Nähere Informationen zum Installieren und Konfigurieren Ihres Browser-Diensts finden Sie in der Dokumentation Ihres Anbieters.
2. Laden Sie Authentic Browser (ein gezippte CAB- oder XPI-Datei) in einen beliebigen Ordner auf dem Server herunter. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss das Paket auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde. **Entpacken Sie diese Datei nicht.** Auf der Website stehen vier Versionen von Authentic Browser zur Verfügung, jede davon in drei Formaten (CAB 32-Bit, CAB 64-Bit und XPI). Informationen zur Auswahl des Dateiformats und der Version finden Sie in den Unterabschnitten dieses Abschnitts.

Hinweis:

Bevor Sie Authentic Browser installieren, vergewissern Sie sich bitte, dass nicht gerade eine frühere Version von Authentic Browser ausgeführt wird, da die neue Version sonst eventuell nicht korrekt registriert wird, was zu einer fehlerhaften Installation führen kann. Um den Fehler zu beheben, registrieren Sie das Plugin durch Ausführen von: `regsvr32 C:\Windows\Downloaded Program Files\AuthenticPlugin.dll`. Beachten Sie bitte, dass Sie zum Ausführen dieses Programms (`regsvr32.exe`) Administratorrechte benötigen.

Dieser Abschnitt enthält die folgenden Unterabschnitte:

- Authentic Browser Datei-Download: Hier wird der Unterschied zwischen dem CAB- und dem XPI-Dateiformat erklärt.
- Authentic Browser-Versionen: Enthält alle entsprechenden Informationen zu den vier Authentic Browser-Versionen.
- Konfigurieren des Browser-Diensts: Hier wird beschrieben, wie Sie bestimmte Dateitypen zur Liste der MIME Types von Interrnet Information Services hinzufügen.

© 2010 Altova GmbH

1 Authentic Browser Datei-Download

CAB-Datei oder XPI-Datei
Das Authentic Browser Plug-in muss auf dem Server als gezippte CAB- oder XPI-Datei gespeichert werden. Ob die CAB- oder die XPI-Datei auf dem Server gespeichert wird, hängt davon ab, welcher Browser auf dem Client verwendet wird, von dem aus die HTML-Seite für das Authentic Plug-in geöffnet wird.

Internet Explorer ver 5.5 oder höher 32-Bit oder 64-Bit	CAB-Datei (. `cab` Dateierweiterung) 32-Bit oder 64-Bit
Firefox	XPI-Datei (. `xpi` Dateierweiterung)

Wenn ein Client beide Browser (Internet Explorer oder Firefox) verwenden könnte, dann sollte sowohl die CAB-Datei als auch die XPI-Datei auf dem Server gespeichert werden. In einem solchen Fall sollten Sie ein Script in der HTML-Seite für das Authentic Plug-in inkludieren, das erkennt, welcher Browser gerade verwendet wird. Die entsprechende Datei (CAB für Internet Explorer; XPI für Firefox) kann dann automatisch vom Server auf den Client heruntergeladen werden. Informationen zu diesem Szenario finden Sie unter HTML-Seite für das Authentic Plug-in | Browser-unabhängig.

CAB-Dateien für Internet Explorer stehen für 32-Bit IE Browser und für 64-Bit IE Browser zur Verfügung. Sie können beide CAB-Dateiarten (32-Bit und 64-Bit) auf dem Server speichern. Die HTML-Seite für das Authentic Plug-in könnte ein Skript enthalten, in dem festgelegt ist, ob es sich beim Browser um eine 32-Bit- oder eine 64-Bit-Version handelt und das die richtige CAB-Datei anschließend herunterlädt. Eine Anleitung zur Erstellung eines solchen Skripts finden Sie im Abschnitt HTML-Seite für das Authentic Plug-in | Browser-unabhängig.

Herunterladen und Speichern der CAB/XPI-Datei
Die CAB-Datei und/oder XPI-Datei muss von der Altova Website heruntergeladen werden und kann in jedem Ordner Ihres Servers gespeichert werden. Wenn Sie die Enterprise Edition von Authentic Browser verwenden, muss das Paket auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde. Beide Dateiformate (CAB und XPI) sind gezippte Formate.

Entpacken Sie diese Dateien nicht. Die Extraktion und Installation der Datei auf dem Client erfolgen automatisch, wenn der Client die HTML-Seite für das Authentic Plug-in zum ersten Mal öffnet. Der Pfad der CAB/XPI-Datei auf dem Server ist in der HTML-Seit für das Authentic Plug-in definiert.

2 Authentic Browser Versionen

Auf der Altova Website stehen vier Versionen des Authentic Browser Plug-in zum Download zur Verfügung. Jede Version ist als CAB-Datei (für Internet Explorer) und als XPI-Datei (für Firefox) verfügbar.

Die verschiedenen Versionen sind weiter unten mit ihren Class IDs (für CAB-Dateien) und MIME Types (für XPI-Dateien) aufgelistet. Sie müssen die entsprechende Class ID bzw. den MIME Type auf der HTML-Seite für das Authentic Plug-in definieren.

- ***CAB-Dateien (separate Dateien für 32-Bit und 64-Bit) und ihre Class IDs***

EN	Trusted	B4628728-E3F0-44a2-BEC8-F838555AE780
EN	Untrusted	A5985EA9-3332-4ddf-AD7F-F6E98BFEAF94
DE	Trusted	9NDDF44A-DFDN-4F47-8EE3-4CBE874584F7
DE	Untrusted	28A640E8-EAEE-4B5D-BEBE-BFA95608NE66

- ***XPI-Dateien und ihre MIME Types***

EN	Trusted	application/x-authentic-scriptable-plugin
EN	Untrusted	application/x-authentic-scriptable-plugin-untrusted
DE	Trusted	application/x-authentic-scriptable-plugin-german
DE	Untrusted	application/x-authentic-scriptable-plugin-untrusted-ge rman

Je nach Anforderung können Sie eine oder mehrere dieser Versionen von der Altova Website herunterladen.

Beachten Sie zu den verschiedenen Authentic Browser Versionen die folgenden Punkte:

- Alle Versionen sind Unicode-Versionen und jede davon bietet vollständige Unterstützung für mehrere Zeichensätze im XML-Dokument. Um Unicode-Versionen verwenden zu können, benötigen Sie Windows XP, Windows Vista oder Windows 7 auf dem Client-Rechner.
- Die Class IDs von .CAB-Dateien für die 32-Bit- und für die 64-Bit IE Browser Version sind identisch und werden in der obigen Tabelle für die veschiedenen EN- und DE-Versionen der Trusted und der Untrusted Version angegeben.
- Die Trusted Version, erlaubt keinen Zugriff auf lokale Dateien und ist daher als "safe for scripting" gekennzeichnet. Sie kann in einem Browser-basierten Szenario verwendet und von jeder Webseite aus ohne Sicherheitswarnungen durch den Client aufgerufen werden.
- Die Untrusted Version ist für die Verwendung im Intranet gedacht oder für die Verwendung der Authentic Browser Edition als ActiveX Control in Ihrer Applikation. Sie bietet Zugriff auf lokale Dateien und ist daher **nicht** als "safe for scripting" gekennzeichnet. Wenn Sie versuchen, diese Version in einem Browser-Fenster zu verwenden, wird der Benutzer um Erlaubnis gefragt.

Hinweise: Informationen zur Installation und Konfiguration Ihres Browser-Diensts (z.B. Internet Information Services von Microsoft) finden Sie in der Dokumentation des jeweiligen Anbieters.

© 2010 Altova GmbH

3 Konfigurieren des Browser-Diensts

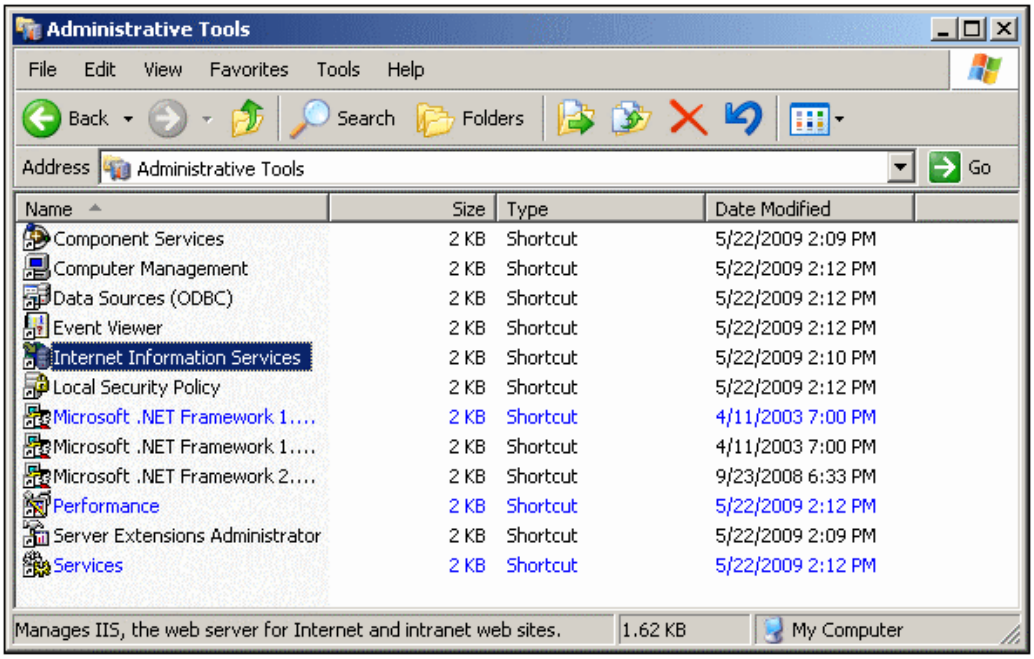
Internet Information Services (IIS) 6 von Microsoft unterstützt nur Dateitypen, die in den MIME Types für die jeweilige Site (Website oder Ordner) definiert sind. Benötigte Dateitypen müssen daher zur Liste der MIME Types für diese Site hinzugefügt werden.

Für die Arbeit mit Authentic Browser werden die folgenden Dateitypen benötigt. Diese müssen hinzugefügt werden:

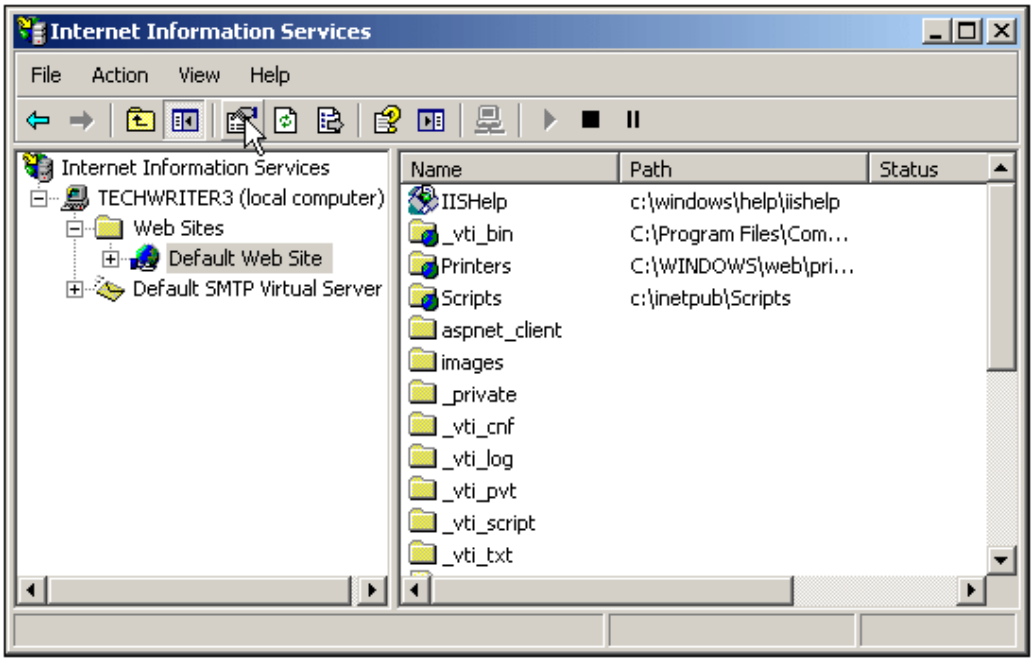
Dateierweiterung	MIME Type	Befehl
xsd	text/plain	
sps	text/plain	
xpi	application/x-xpinstall	Für Firefox. Wird für IE nicht benötigt.

Hinzufügen von MIME Types für eine Site in Internet Information Services
Um auf einem Windows XP-Betriebssystem zur Liste der MIME Types für eine bestimmte Website einen MIME Type hinzuzufügen, gehen Sie folgendermaßen vor. Die Vorgehensweise ist auch auf anderen unterstützten Systemen (Windows Vista und Windows 7) ähnlich.

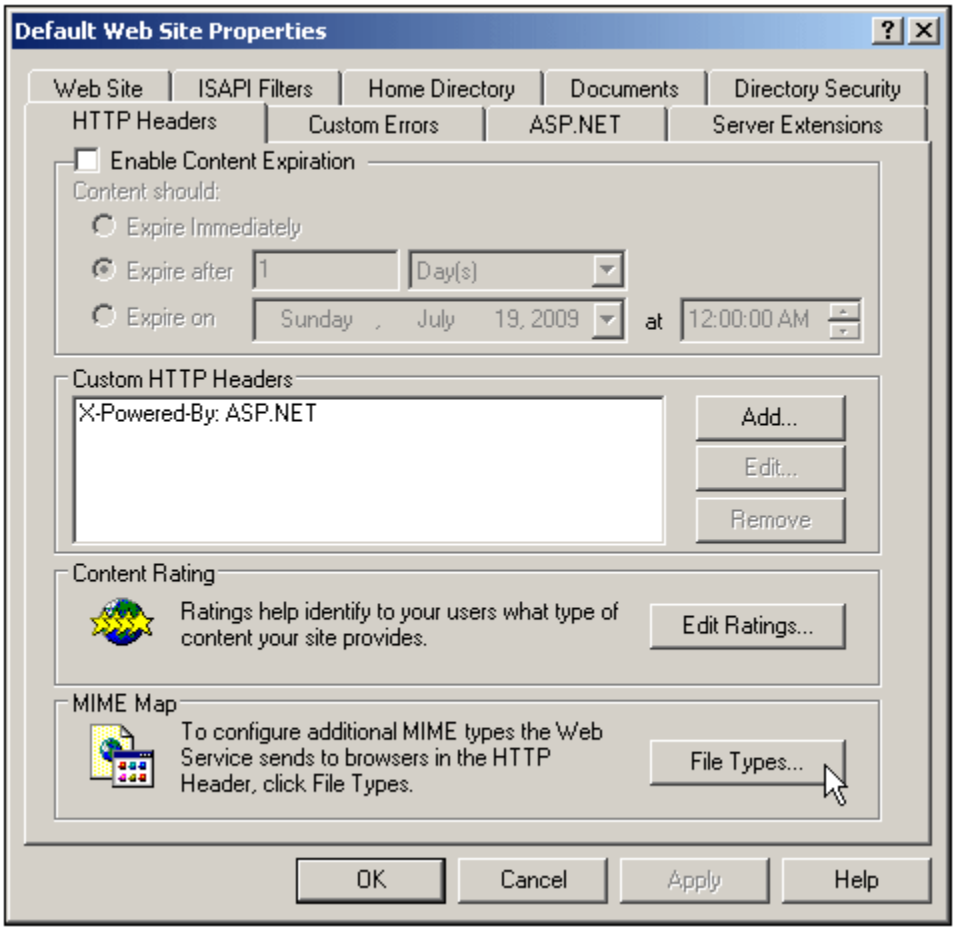
1. Öffnen Sie die Systemsteuerung und doppelklicken Sie auf "Verwaltung".
2. Doppelklicken Sie anschließend im angezeigten Ordner auf Internet Information Services (*Abbildung unten*).


Administrative Tools
File Edit View Favorites Tools Help
Back Search Folders
Address Administrative Tools
Go
Name Size Type Date Modified
Component Services 2 KB Shortcut 5/22/2009 2:09 PM
Computer Management 2 KB Shortcut 5/22/2009 2:12 PM
Data Sources (ODBC) 2 KB Shortcut 5/22/2009 2:12 PM
Event Viewer 2 KB Shortcut 5/22/2009 2:12 PM
Internet Information Services 2 KB Shortcut 5/22/2009 2:10 PM
Local Security Policy 2 KB Shortcut 5/22/2009 2:12 PM
Microsoft .NET Framework 1.... 2 KB Shortcut 4/11/2003 7:00 PM
Microsoft .NET Framework 1.... 2 KB Shortcut 4/11/2003 7:00 PM
Microsoft .NET Framework 2.... 2 KB Shortcut 9/23/2008 6:33 PM
Performance 2 KB Shortcut 5/22/2009 2:12 PM
Server Extensions Administrator 2 KB Shortcut 5/22/2009 2:09 PM
Services 2 KB Shortcut 5/22/2009 2:12 PM
Manages IIS, the web server for Internet and intranet web sites.
1.62 KB
My Computer

3. Wählen Sie im Ordner "Internet Information Services (IIS)" (*Abbildung unten*) zuerst im Ordnerbereich die gewünschte Site aus (Website oder Ordner) und klicken Sie anschließend auf die Schaltfläche **Eigenschaften** (*in der Abbildung unter dem Cursor*) oder im Kontextmenü, das durch Rechtsklick aufgerufen wird, auf den Befehl **Eigenschaften**.

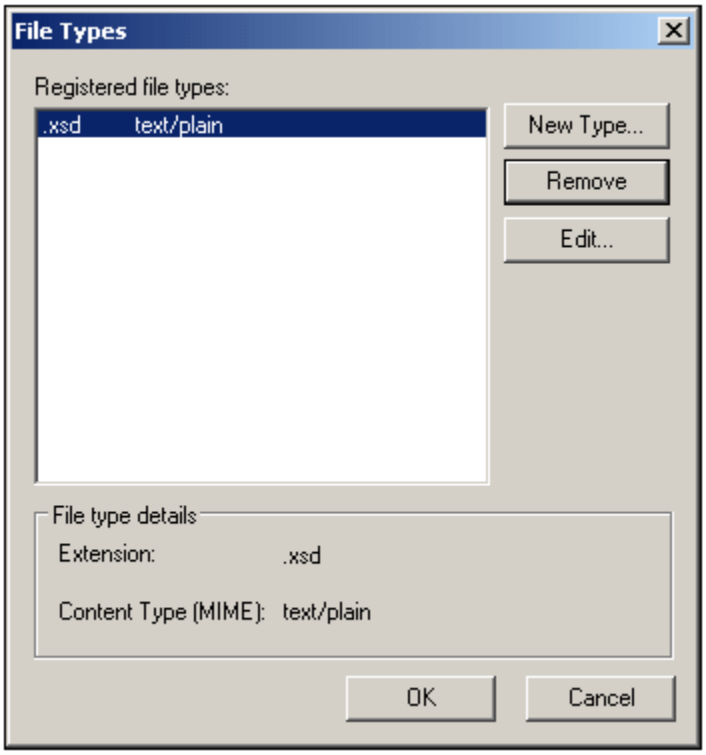


4. Klicken Sie im Dialogfeld "Eigenschaften" auf dem Register "HTTP Headers" i Bereich "MIME-Zuordnungen" auf die Schaltfläche **Dateitypen** (*siehe Abbildung unten*).

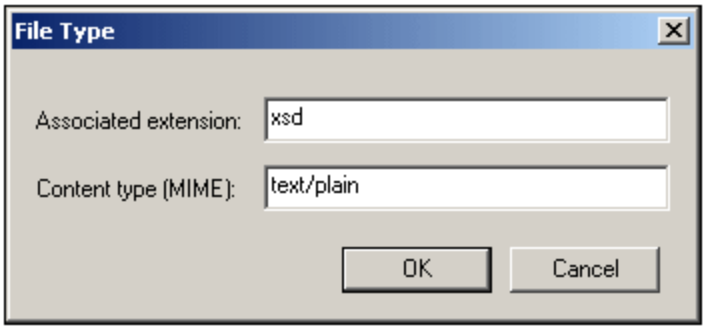

Default Web Site Properties
Web Site ISAPI Filters Home Directory Documents Directory Security
HTTP Headers Custom Errors ASP.NET Server Extensions
Enable Content Expiration
Content should:
Expire Immediately
Expire after 1 Day(s)
Expire on Sunday , July 19, 2009 at 12:00:00 AM
Custom HTTP Headers
X-Powered-By: ASP.NET
Add...
Edit...
Remove
Content Rating
Ratings help identify to your users what type of content your site provides.
Edit Ratings...
MIME Map
To configure additional MIME types the Web Service sends to browsers in the HTTP Header, click File Types.
File Types...
OK Cancel Apply Help

 © 2010 Altova GmbH

5. Klicken Sie im Dialogfeld "Dateitypen" (*Abbildung unten*) auf die Schaltfläche **Neuer Typ...**.



6. Geben Sie im daraufhin angezeigten Dialogfeld die gewünschte Erweiterung und ihren MIME Type ein. Informationen zu den erforderlichen Dateitypen und ihren entsprechenden MIME Types finden Sie in der Tabelle oben.



7. Bestätigen Sie Ihre Auswahl mit **OK**.

Eine Beschreibung, wie man über Web-DAV Fernzugriff auf einen Ordner auf einem internen Server einrichten kann, finden Sie in diesem Windows IT Pro-Artikel.

Kapitel 5

Anmerkungen zum Einrichten des SPS

Anmerkungen zum Einrichten des SPS

Damit Authentic Browser ordnungsgemäß funktioniert, müssen Sie sicher stellen, dass in der SPS-Datei **ein Schema als Hauptschema zugewiesen wurde**. Ausführliche Informationen zum Erstellen von StyleVision Power Stylesheets (SPSs) finden Sie im Benutzerhandbuch zu Altova StyleVision.

Datenbank-basierte SPS
Falls Sie Authentic Browser verwenden wollen, um Datenbanken (DB) über ein Datenbank-basiertes StyleVision Power Stylesheet (SPS) aufzurufen oder zu editieren, müssen Sie die folgenden Einstellungen vornehmen, um sicherzustellen, dass der Client eine Verbindung zur Datenbank aufbauen kann.

Connection Informationen im SPS: Alle für die Herstellung der Verbindung zur Datenbank erforderlichen Informationen sind in einem Connection String im SPS gespeichert. Der Connection String im SPS wird bei der Erstellung des StyleVision Power Stylesheet in StyleVision erstellt. Das Verfahren zur Herstellung einer Verbindung zu MS Access Datenbank unterscheidet sich von den für andere Datenbanken verwendeten Verfahren. Für andere Datenbanken wird eine ADO-Verbindung verwendet. Im Folgenden sind die Einstellungen beschrieben, die Sie für diese beiden Verbindungsarten vornehmen müssen.

- **Für MS Access-Datenbanken:** Im Connection String für MS Access Datenbanken muss ein UNC-Pfad verwendet werden, damit der Client eine Verbindung zur Datenbank aufbauen kann. Dieser UNC-Pfad wird beim Erstellen des StyleVision Power Stylesheet in StyleVision definiert. Sie müssen allerdings sicherstellen, dass der Ordner, in dem die Datenbank gespeichert ist (oder ein übergeordneter Ordner) freigegeben wurde. (In Windows XP können Sie diese Einstellungen durch Rechtsklick auf den Ordner und Auswahl des Befehls "**Freigabe und Sicherheit...**" vornehmen). Außerdem müssen Sie auf diesem Rechner die erweiterte Einstellung zur Dateifreigabe aktivieren (Klicken Sie auf **Arbeitsplatz | Extras | Ordneroptionen** und deaktivieren Sie das Kontrollkästchen "einfache Datenfreigabe verwenden"). Auf den Clients sind keine speziellen Einstellungen erforderlich.

  Hinweis: Das Format für den im Connection String angegebenen UNC-Pfad ist: `\\servername\sharename\path\file.mdb`, wobei `servername` der Name des Servers ist, `sharename` der Name des freigegebenen Ordners (der in den Freigabeeinstellungen für den jeweiligen Ordner auf dem Server definiert wurde), `path` der Pfad zur Datenbank und `file.mdb` der Name der MS Access Datenbank im freigegebenen Ordner, bzw. in einem Unterordner dieses Ordners.

- **Für ADO-Verbindungen:** Der ADO Connection String im SPS wird beim Erstellen des SPS in StyleVision definiert. Er enthält die erforderlichen Connection-Informationen sowie die Sicherheitsinformationen. Sie müssen sicherstellen, dass der beim Testen des Connection String in StyleVision verwendete Treiber auch auf allen Client-Rechnern vorhanden ist, auf denen Authentic Browser installiert ist, damit das SPS eine Verbindung zwischen Datenbank und Client aufbauen kann. Beachten Sie außerdem, dass ADO-Datenbankverbindungen nur mit lokalen Dateipfaden und nicht mit `http://` URLs funktionieren.

© 2010 Altova GmbH

Kapitel 6

HTML-Seite für das Authentic Plug-in

HTML-Seite für das Authentic Plug-in

Die HTML-Seite für das Authentic Plug-in führt die folgenden wichtigen Funktionen aus:

1. Wenn die HTML-Seite für das Authentic Plug-in das erste Mal auf dem Client-Rechner geöffnet wird, wird das Authentic Browser Plug-in aufgrund des Codes in der HTML-Seite vom Server auf den Client heruntergeladen und dort installiert. Dieser Code muss korrekt sein, damit die richtige Datei auf dem Server gefunden wird. Das Format der Datei, die heruntergeladen wird (CAB oder XPI) ist abhängig vom Browser des Client. Der Code unterscheidet sich auch, weil Internet Explorer und Firefox Code unterschiedlich verarbeiten.
2. Der Code in der HTML-Seite richtet die Authentic-Ansicht im Browser-Fenster ein, dabei werden auch die Größe der Authentic-Ansicht und die Pfade zu der XML-, XSD- und SPS-Datei konfiguriert.
3. Die zu bearbeitende XML-Datei sowie die Schema-Datei und das SPS, auf dem die XML-Datei basiert, werden definiert.
4. Die HTML-Seite enthält in den HTML `SCRIPT` Elementen Definitionen für Subroutinen und die Behandlung von Events. So kann z.B. festgelegt werden, welche Aktion ausgeführt werden soll, wenn man auf der HTML-Seite auf eine Schaltfläche klickt. Beispiele dazu finden Sie unter Internet Explorer Beispiel 1: Einfach und Firefox Beispiel 1: Einfach.

Anmerkung: Wenn Sie die **Enterprise Edition von Authentic Browser** verwenden, muss die HTML-Seite auf dem Server gespeichert werden, für den die Enterprise Lizenz registriert wurde.

In den Unterabschnitten dieses Abschnitts wird beschrieben, wie die oben aufgelisteten Funktionen auf der HTML-Seite implementiert werden sollen. Diese Abschnitten sind aufgrund der unterschiedlichen Methoden, wie die Authentic Browser DLL für bestimmte Browser heruntergeladen wird, auf der ersten Ebene nach Browser-Art gegliedert:

- Lizenzierung für die Enterprise Edition enthält eine Beschreibung zur Lizenzierung für die Enterprise Edition von Authentic Browser
- Internet Explorer enthält eine Beschreibung, wie man eine CAB-Datei (mit der Erweiterung `.cab`) herunter über ein HTML `OBJECT`-Element herunterlädt.
- Firefox: Enthält eine Beschreibung, wie man eine XPI-Datei (Erweiterung `.xpi`) über ein HTML `EMBED`-Element herunterlädt.
- Unabhängige Browser: Enthält eine Beschreibung, wie man feststellt, welche Browser-Art den Request sendet und wie man die richtige Plug-In-Version für diesen Browser herunterlädt.

In diesen Unterabschnitten finden Sie Beschreibungen und Beispiele zur Verwendung der HTML-Elemente `OBJECT` , `EMBED` und `SCRIPT`. Außerdem finden Sie Beispiel für vollständige HTML-Seiten, die das Authentic Browser Plug-in aufrufen. Nähere Informationen zu einzelnen Objekten finden Sie im Abschnitt Benutzerreferenz unter der jeweiligen Objektbeschreibung.

© 2010 Altova GmbH

1 Lizenzierung für die Enterprise Edition

Authentic Browser ist in zwei Versionen (Editions) verfügbar:

1. einer **Enterprise Edition**, die komplexe SPS-Funktionen unterstützt und für die eine Lizenz erforderlich ist
2. einer **Community Edition**, für die keine Lizenz benötigt wird.

Die Lizenz für die Authentic Browser Enterprise Edition können Sie auf der Altova Website erwerben.

Übersicht über das Einrichten der Lizenz für die Enterprise Edition
Unten finden Sie eine kurze Übersicht, welche Schritte erforderlich sind, um die Lizenzinformationen für die Authentic Browser Enterprise Edition korrekt einzurichten.

- Für die Authentic Browser Enterprise Edition werden die folgenden gültigen Lizenzinformationen benötigt: (i) der ***Server-Name,*** für den die Lizenz gilt; (ii) der ***Name des Unternehmens,*** für das die Lizenz registriert wurde und (iii) der ***Lizenzschlüssel***. Diese Informationen werden Ihnen in der Lizenz-E-Mail zugesandt, die Sie nach Erwerb Ihrer Authentic Browser Enterprise Edition-Lizenz erhalten.
- Die Lizenzinformationen müssen auf der HTML-Seite für das Authentic Plug-in vorhanden sein. Wie dies auf der HTML-Seite erfolgt, wird unten beschrieben. (Beachten Sie: Es gibt keine spezifische Lizenzschlüsseldatei, die von Authentic Browser referenziert wird.)
- Wenn die auf der HTML-Seite für das Authentic Plug-in eingetragenen Informationen gültig sind, so werden die Funktionalitäten der Enterprise Edition in der Authentic Browser-Datei freigegeben. Andernfalls stehen Ihnen nur die Funktionen der Community Edition zur Verfügung.
- Die HTML-Seite für das Authentic Plug-in muss auf dem Server gespeichert sein, für den die Lizenz gilt.

So geben Sie die Lizenzinformationen ein
Die drei Lizenzschlüsselparameter (Servername, Firmenname und Lizenzschlüssel) können auf mehrere Arten auf der HTML-Seite für das Authentic Plug-in eingetragen werden:

- Als Parameterwerte der Elemente `OBJECT` oder `EMBED`. Eine Anleitung dazu finden Sie in den entsprechenden Abschnitten zur HTML-Seite für Internet Explorer und Firefox.
- Wenn die HTML-Seite Browser-unabhängig sein soll, können die Lizenzparameter wie in der Codeliste im Browser-unabhängigen Beispiel gezeigt, registriert werden.
- Der Lizenzparameter kann auch direkt im Objekt gesetzt werden, wie in der Codeliste, in der die Codeliste im Browser-unabhängigen Beispiel adaptiert wurde, unten gezeigt wird.

```
<SCRIPT LANGUAGE=javascript>
// event subscription if running on Firefox
if ( isFirefoxOnWindows() )
{
	objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage,
false);
}
</SCRIPT>
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
// event subscription if running on Internet Explorer
if ( isIEOnWindows() )
{
	InitAuthenticPluginPage();
}
</SCRIPT>

<SCRIPT type="text/javascript" LANGUAGE="javascript" >
```

```
        function InitAuthenticPluginPage( )
        {
        var serverstr='DevAuthBrowTest';
        var basedir='Authentic/';
        objPlugIn.LicServer = 'DevAuthBrowTest';
        objPlugIn.LicCompany = 'Altova';
        objPlugIn.LicKey = 'XXXXXXXXXX';
        objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
        objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
        objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
        objPlugIn.StartEditing();
        }
        </SCRIPT>
```


© 2010 Altova GmbH

2 Internet Explorer

Wenn die HTML-Seite für das Authentic Plug-in in Internet Explorer (32-Bit oder 64-Bit) geöffnet werden soll, muss sie die folgenden Elemente enthalten:

- ein HTML `OBJECT`-Element, das (i) die richtige DLL für das Authentic Plug-In (32-Bit oder 64-Bit) vom Server auf den Client lädt und (ii) die Größe des Authentic-Fensters im Browser des Client definiert. Das `OBJECT`-Element enthält den Pfad zum Authentic Browser Plug-in. **Beachten Sie, dass das Authentic Plug-in in einer Version für die 32-Bit- und in einer Version für die 64-Bit-Version von Internet Explorer zur Verfügung steht, daher muss die richtige Authentic Browser Plug-in-Version (`.cab` Datei) auf den Client heruntergeladen werden.** Beispielcode, der die X-Bit Verson von Internet Explorer automatisch überprüft und das richtige Authentic Plug-in herunterlädt, finden Sie im Abschnitt Browser-unabhängiges Beispiel.
- eines oder mehrere HTML `SCRIPT`-Elemente. Ein `SCRIPT`-Element definiert die verschiedenen Subroutinen, die in der HTML-Seite und bei der Ereignisverarbeitung verwendet werden. Mit Hilfe eines `SCRIPT`-Elements können Sie definieren, welches XML-Dokument bearbeitet werden soll und welches XML-Schema und welche SPS-Datei dem XML-Dokument zugrunde gelegt werden soll.

Dieser Abschnitt ist in die folgenden Unterabschnitte gegliedert:

- Das OBJECT-Element: Hier wird beschrieben, wie das HTML `OBJECT` Element in einer HTML-Seite für das Authentic Plug-in verwendet wird.
- Das SCRIPT-Element: Hier wird beschrieben, wie HTML `SCRIPT` Elemente in einer HTML-Seite für das Authentic Plug-in verwendet werden.
- Beispiele für komplette HTML-Seiten: IE Beispiel 1, einfach und IE Beispiel 2: Tabelle sortieren.

Informationen zu einzelnen Objekten finden Sie im Abschnitt "Benutzerreferenz" unter der entsprechenden Object-Beschreibung.

Hinweis: Das Authentic Browser Plug-in wird für **Internet Explorer 5.5 oder höher** unterstützt.

2.1 Das OBJECT-Element

Das `OBJECT`-Element hat die folgenden Funktionen:

- Es gibt dem Authentic Browser Plug-In einen Namen (über das `id`-Attribut).
- Es wählt aus, welche Version des Authentic Browser Plug-In verwendet werden soll (über den Wert des `classid`-Attributs).
- Es definiert eine a `.cab` Datei und eine Versionsnummer (`codebase`-Attribut). Die `.cab`-Datei enthält eine DLL, die ein COM-Objekt (das Authentic Browser Plug-In) mit einer ID registriert, die der Wert des `classid`-Attributs ist. Normalerweise wird sowohl die 32-Bit- als auch die 64-Bit-Version des Authentic Browser Plug-in auf dem Server gespeichert. Jede Version wird durch einen unterschiedlichen Dateinamen gekennzeichnet. Das `codebase` Attribut definiert den Dateinamen der erforderlichen `.CAB` Datei (*siehe Codeliste weiter unten*). Die Class IDs für die 32-Bit und die 64-Bit `.CAB` Dateien sind identisch und werden in der Tabelle für die verschiedenen EN/DE-Versionen der Trusted/Untrusted Version aufgelistet.

  Dateiname für die 32-Bit Version: `AuthenticBrowserEdition.CAB`
  Dateiname für die 64-Bit Version: `AuthenticBrowserEdition_x64.CAB`

- Es definiert (über das `style`-Attribut) die Größe des Authentic-Fensters im Browser des Client .
- Es kann eine beliebige Anzahl von Parametern definieren.

Hier sehen Sie ein Beispiel für ein HTML `OBJECT`-Element. Es wählt die Trusted Unicode-Version aus (über den im `classid`-Attribut definierten Wert) und legt als Größe des Authentic-Fensters im Browser des Client 600 x 500 Pixel fest. Im Folgenden sind alle Attribute und Parameter beschrieben, die dem `OBJECT`-Element zur Verfügung stehen. (Beachten Sie, dass die unten angegebene Versionsnummer möglicherweise nicht die Nummer der aktuellen Version ist. Nähere Informationen dazu siehe codebase.)

Für das 32-Bit Authentic Browser Plug-in:

```
<OBJECT id="objPlugIn" style="WIDTH:600px; HEIGHT:500px"
   codeBase="
http://yourserver/cabfiles/AuthenticBrowserEdition.CAB#Version=N2,2,0,0"
   classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780">
   <PARAM NAME="XMLDataURL" VALUE="http://yourserver/OrgChart.xml">
   <PARAM NAME="SPSDataURL" VALUE="http://yourserver/OrgChart.sps">
   <PARAM NAME="SchemaDataURL" VALUE="http://yourserver/OrgChart.xsd">
</OBJECT>
```

Für das 64-Bit Authentic Browser Plug-in:

```
<OBJECT id="objPlugIn" style="WIDTH:600px; HEIGHT:500px"
   codeBase="
http://yourserver/cabfiles/AuthenticBrowserEdition_x64.CAB#Version=N2,2,0,0"
   classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780">
   <PARAM NAME="XMLDataURL" VALUE="http://yourserver/OrgChart.xml">
   <PARAM NAME="SPSDataURL" VALUE="http://yourserver/OrgChart.sps">
   <PARAM NAME="SchemaDataURL" VALUE="http://yourserver/OrgChart.xsd">
</OBJECT>
```

`id`
Der Wert des `id`-Attributs wird als Name der Authentic Browser Plug-In Objekte verwendet, wenn diese in Skripts verwendet werden. So wird z.B. mit `objPlugIn.SchemaLoadObject.URL` das Objekt aufgerufen, das die Schema-Datei lädt. Nähere Informationen dazu siehe "Das SCRIPT-Element".

`style`
Dies ist das normale HTML `style`-Attribut. Es dient zur Angabe der Größe des Authentic

-Fensters im Browser des Client.

codebase
Das codebase-Attribut gibt den Pfad zur .CAB-Datei an. Beachten Sie, dass die .CAB-Datei für das 32-Bit Authentic Browser Plug-in und die .CAB-Datei für das 64-Bit Authentic Browser Plug-in nicht dieselben sind, nämlich: AuthenticBrowserEdition.CAB und AuthenticBrowserEdition_x64.CAB.

Der Wert der optionalen #Version-Erweiterung gibt die Versionsnummer der Komponente an, die derzeit auf dem Server zur Verfügung steht. Falls der Client eine frühere Version verwendet und im codebase Attribut eine neuere Version definiert ist, wird die neuere Version vom Server installiert. Wenn die #Version-Erweiterung nicht angegeben wurde, erfolgt die Aktualisierung erst, wenn die Komponente manuell vom Client entfernt wurde. Die aktuelle **Versionsnummer der Komponente ist in den Eigenschaften der .dll-Datei der .CAB-Datei der Komponente aufgelistet.**

classid
Die Class IDs für die 32-Bit und die 64-Bit .CAB Datei sind identisch und sind die in der nachfolgenden Liste angeführten IDs für die EN/DE- und die Trusted/Untrusted Version.

Ab Version 5.0 des Browser Plug-In ist der classid-Wert für Unicode-Versionen ab Version 5.0 ein anderer als der früherer Unicode-Versionen. Wenn Sie daher die Unicode .CAB-Datei auf Ihrem Server von einer Version vor Version 5.0 aktualisieren, stellen Sie sicher, dass Sie die classid -Werte in Ihren HTML-Dateien ändern. Beachten Sie: Wenn eine neue .CAB-Datei auf dem Server dieselbe CLSID hat wie eine auf dem Client bereits installierte .CAB-Datei, wird die alte .CAB-Datei auf dem Client nicht automatisch durch die neue ersetzt. Die müssen die zuvor installierte .CAB-Datei entfernen, bevor Sie die neue .CAB-Datei herunterladen. Jede Sprachversion hat einen anderen CLSID-Wert.

Bedenken Sie bei der Auswahl der Version, die auf den Client heruntergeladen werden soll, dass die Unicode-Version auf älteren Betriebssysteme möglicherweise nicht korrekt installiert werden kann.

Im Folgenden finden Sie die CLSID-Werte der aktuellen Versionen (Deutsch (DE) und Englisch (EN)):

DE Trusted Unicode:
clsid:9NDDF44A-DFDN-4F47-8EE3-4CBE874584F7

DE Untrusted Unicode:
clsid: 28A640E8-EAEE-4B5D-BEBE-BFA95608NE66

EN Trusted Unicode:
clsid: B4628728-E3F0-44a2-BEC8-F838555AE780

EN Untrusted Unicode:
clsid: A5985EA9-3332-4ddf-AD7F-F6E98BFEAF94

Parameter
Es können beliebig viele der folgenden Parameter verwendet werden.

LicServer
Der Name des Servers, für den die Authentic Browser Enterprise Edition-Lizenz gültig ist. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

LicKey

Der Lizenzschlüssel, der für die Authentic Browser Enterprise Edition benötigt wird. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

LicCompany
Der Name des Unternehmens, für das die Verwendung der Authentic Browser Enterprise Edition lizenziert wurde. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

XMLDataURL
Eine absolute URL, die den Speicherort der zu editierenden XML-Datei angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

XMLDataSaveURL
Eine absolute URL, die den Ordner angibt, in dem die XML-Datei gespeichert werden soll. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

SPSDataURL
Eine absolute URL, die den Pfad zum StyleVision Power Stylesheet (`.sps`-Datei) angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

SchemaDataURL
Eine absolute URL, die den Pfad zur verknüpften Schema-Datei angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

TextStateBmpURL
Der Ordner, in dem Bitmap-Grafiken für Textstatus-Symbole gespeichert werden sollen.

TextStateToolbarLine
Die Zeile der Symbolleiste, in der Textstatus-Symbole platziert werden sollen. Die Standardeinstellung ist 1.

AutoHideUnusedCommandGroups
Gibt an, ob Gruppen nicht verwendeter Symbolleistenbefehle ausgeblendet werden sollen. Die Standardeinstellung ist True.

ToolbarsEnabled
Gibt die allgemeine Unterstützung für Symbolleisten an. Die Standardeinstellung ist True.

ToolbarTooltipsEnabled
Gibt an, ob Tooltipps aktiviert oder deaktiviert sind.

HideSaveButton
Bei Einstellung "True" wird die standardmäßig eingeblendete Schaltfläche "Speichern" aus der Authentic-Symbolleiste entfernt.

BaseURL
Gibt die Basis-URL für relative Pfade an.

SaveButtonUsePOST
Bei Einstellung "True" wird beim Speichern des Dokuments der Befehl HTTP POST verwendet, anstelle von PUT.

EntryHelpersEnabled
Bei Einstellung "True" werden die Authentic-Eingabehilfen eingeblendet.

EntryHelperSize
Breite der Eingabehilfefenster in Pixeln.

 © 2010 Altova GmbH

EntryHelperAlignment
Definiert die Position der Eingabehilfen im Dokumentenfenster.

- 0 = Symbolleiste am oberen Rand des Dokuments anzeigen
- N = Symbolleiste am linken Rand des Dokuments anzeigen
- 2 = Symbolleiste am unteren Rand des Dokuments anzeigen
- 3 = Symbolleiste am rechten Rand des Dokuments anzeigen

EntryHelperWindows
Gibt an, welche der Eingabehilfenfenster eingeblendet werden sollen.

- N = Elemente
- 2 = Attribute
- 4 = Entities

Zulässig ist jede Kombination (bit-check)

SaveButtonAutoEnable
Siehe Authentic.SaveButtonAutoEnable

LoaderSettingsFileURL
Gibt die URL der `LoaderSettingsFile` für die Paketverwaltung an.

Die folgenden Parameter werden nicht mehr verwendet:
Ab Version 6.0.3.1 von Authentic Browser sind keine Lizenzen mehr erforderlich.

LicServer
Der Name des Servers, auf dem sich der Lizenzkey befindet (falls der Lizenzkey über eine Datei zur Verfügung gestellt wird).

LicFile
Der Name der Datei, in der der Lizenzkey gespeichert ist.

LicKey
Der Lizenzkey zur Überprüfung der Verwendung der Authentic Browser Edition.

LicCompany
Der Name des Unternehmens (zur Überprüfung der Verwendung der Authentic Browser Edition).

LicUser
Der Name des Benutzers (zur Überprüfung der Verwendung der Authentic Browser Edition).

2.2 Das SCRIPT-Element

Die SCRIPT-Elemente definieren die Ereignishandler und Subroutinen, die von innerhalb der HTML-Datei aus aufgerufen werden können.

Im Folgenden sehen Sie ein Beispiel für ein Skript zur Ereignisbehandlung:

```
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
      objPlugIn.SchemaLoadObject.URL  = "http://yourserver/OrgChart.xsd"
      objPlugIn.XMLDataLoadObject.URL  = "http://yourserver/OrgChart.xml"
      objPlugIn.DesignDataLoadObject.URL  = "http://yourserver/OrgChart.sps"
      objPlugIn.StartEditing
</SCRIPT>
```

Hier sehen Sie ein Beispiel für ein Skript mit Subroutinen:

```
<SCRIPT ID=clientEventHandlers LANGUAGE=vbscript>
   Sub BtnOnClick
      objPlugIn.SchemaLoadObject.URL  = "http://yourserver/OrgChart.xsd"
      objPlugIn.XMLDataLoadObject.URL  = "http://yourserver/OrgChart.xml"
      objPlugIn.DesignDataLoadObject.URL  = "http://yourserver/OrgChart.sps"
      objPlugIn.StartEditing
   End Sub
   Sub OnClickFind
      objPlugIn.FindDialog
   End Sub
   Sub BtnOnTestProp
      If objPlugIn.IsRowInsertEnabled Then
         msgbox "true"
      Else
         msgbox "false"
      End If
   End Sub
</SCRIPT>
```

LAkGUAGE
Das Authentic Browser Plug-In wurde mit JavaScript and VBScript getestet.

Ereignisbehandlung
Der Wert des ID-Attributs des OBJECT-Elements im HTML-Textkörper wird als der Wert des FOR-Attributs definiert. Authentic Browser Plug-In-Objekte, die aufgerufen werden, müssen einen Namen haben, der diesem Wert entspricht.

Eine Liste von Ereignissen finden Sie auch unter Ereignisse: Referenz.

Subroutinen
Für jedes Ereignis, das Sie in der HTML-Datei definieren möchten, können Subroutinen erstellt werden. Der Objektname für das Authentic Browser Plug-In muss mit dem Wert des ID -Attributs des OBJECT-Elements im HTML-Textkörper identisch sein. Im oben gezeigten Beispiel ist das Präfix objPlugIn. Dieses Präfix muss der Wert des ID-Attributs des OBJECT -Elements sein.

Die im Authentic Browser Plug-In verfügbaren Methoden, Eigenschaften und Sub-Objekte werden im Referenzteil dieser Dokumentation beschrieben.

© 2010 Altova GmbH

2.3 IE Beispiel 1: Einfach

Der unten gezeigte HTML-Code generiert eine Seite, die die folgenden Funktionen hat:

- Sie installiert die Trusted Unicode-Version von Authentic Browser am Client, falls dies nicht bereits geschehen ist.
- Innerhalb des body-Tags ist ein Fenster von der Größe 600px mal 500px definiert, in das Authentic Browser geladen wird.
- Unterhalb des Authentic Browser-Fensters befindet sich eine Reihe mit vier Schaltflächen
- Die Authentic-Ansicht von `OrgChart.xml` ist geladen.
- Die Schaltflächen **Suchen** und **Ersetzen** dienen zum Aufrufen des Such- bzw. Ersetzungsdialogfelds.
- Die Schaltfläche **Speichern** dient zum Speichern der Änderungen in einer Datei mit dem Namen `SaveFile.xml`, die im Root-Verzeichnis des Servers abgelegt ist.
- Die Schaltfläche **Eigenschaft testen** dient zum Testen einer einfachen Eigenschaft.

Wenn diese HTML-Seite auf dem Client geöffnet wird, kann der Benutzer die XML-Datei `OrgChart.xml` bearbeiten und die bearbeitete Datei als `SaveFile.xml` speichern.

Anhand dieser einfachen HTML-Seite können Sie testen, ob Authentic Browser korrekt funktioniert. Stellen Sie dabei sicher, dass Sie zur Angabe der `CAB`-Datei, der `xsd`-, `xml`- und der `sps`-Datei und aller anderen auf dem Server gespeicherten Ressourcen die richtigen URLs verwenden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code. Sie können dieses Beispiel erweitern und ändern, um komplexere Lösungen mit Hilfe von Authentic Browser zu erstellen. Nähere Informationen dazu finden Sie auch unter dem OBJECT-Element.

```
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
    <title>Minimal XMLSpyDocEditPlugIn page</title>

    <!-- Script for handling the ControlInitialized event -->
    <SCRIPT LANGUAGE="javascript" FOR="objPlugIn" EVENT="ControlInitialized">
         objPlugIn.SchemaLoadObject.URL = "http://yourserver/OrgChart.xsd"
         objPlugIn.XMLDataLoadObject.URL = "http://yourserver/OrgChart.xml"
         objPlugIn.DesignDataLoadObject.URL = "http://yourserver/OrgChart.sps"
         objPlugIn.StartEditing()
    </SCRIPT>

    <!-- Script with subroutines -->
    <SCRIPT ID=clientEventHandlers LANGUAGE=vbscript>
      Sub OnClickFind
          objPlugIn.FindDialog
      End Sub

      Sub OnClickReplace
          objPlugIn.ReplaceDialog
      End Sub

      Sub BtnOnSave
          objPlugIn.XMLDataSaveUrl = "http://yourserver/SaveFile.xml"
          objPlugIn.Save
      End Sub

      Sub BtnOnTestProp
          If objPlugIn.IsRowInsertEnabled Then
             msgbox "true"
          Else
             msgbox "false"
          End If
      End Sub
    </SCRIPT>
  </head>
```

```
  <body>
    <!-- Object element has id with value that must be used -->
    <!-- as name of Authentic Browser Plug-in objects -->
    <!-- Classid selects the Trusted Unicode version -->
    <OBJECT id="objPlugIn"
    <!-- CodeBase selects 32-bit CAB file (AuthenticBrowserEdition.CAB) -->
    <!-- or 64-bit CAB file (AuthenticBrowserEdition_x64.CAB) -->
      CodeBase="http://yourserver/AuthenticBrowserEdition.CAB#Version=N2,2,0,0"
    <!-- Class Id for 32-bit and 64-bit CAB files is the same -->
      Classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780" width="600" height="500">
    </OBJECT>
    <p>
      <input type="button" value="Find" name="B4" onclick="OnClickFind()">
      <input type="button" value="Replace" name="B5" onclick="OnClickReplace()">
      <input type="button" value="Save" name="B6" onclick="BtnOnSave()">
      <input type="button" value="Test property" name="B7" onclick="BtnOnTestProp">
    </p>
  </body>
</html>
```


© 2010 Altova GmbH

2.4 IE Beispiel 2: Tabelle sortieren

Dies ist eine HTML-Beispieldatei mit einem eingebetteten JavaScript. Für das Beispiel muss das Authentic Browser Plug-In (CAB-Datei und Lizenzdatei) auf Ihrem Computer installiert werden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code.

Der Code zeigt an:

- wie das Browser Plug-In aufgerufen werden soll. Passen Sie den Code bitte an und geben Sie den Pfad zu Ihrer CAB-Datei und dem Class Identifier Ihrer Browser Plug-In-Version an (trusted oder untrusted).
- wie eine Datei in das Browser Plug-In geladen werden soll. Passen Sie bitte den Code an und geben Sie den Pfad zu Ihrem Beispieldokument an.
- wie Schaltflächen für einfache Cursor-Positionierungen implementiert werden.
- wie komplexere Befehle wie z.B. die Sortierung von Tabellen implementiert werden.
- wie das `SelectionChanged`-Ereignis verwendet wird.

Nähere Informationen dazu finden Sie auch unter dem OBJECT-Element.

```
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
      <title>test page For Authentic Browser Plug-in</title>

      <SCRIPT LANGUAGE="javascript" For="objPlugIn" EVENT="ControlInitialized">
         var strSampleRoot = "http://myRoot/myPath/myDocBaseName";
         objPlugIn.SchemaLoadObject.URL = strSampleRoot + ".xsd";
         objPlugIn.XMLDataLoadObject.URL = strSampleRoot + ".xml";
         objPlugIn.DesignDataLoadObject.URL = strSampleRoot + ".sps";
         objPlugIn.StartEditing();
      </SCRIPT>

      <SCRIPT ID="clientEventHandlers" LANGUAGE="javascript">
         var objCurrentRange = Null;

         Function BtnDocumentBegin() { objPlugIn.AuthenticView.DocumentBegin.Select();
}
         Function BtnDocumentEnd() { objPlugIn.AuthenticView.DocumentEnd.Select(); }
         Function BtnWholeDocument() { objPlugIn.AuthenticView.WholeDocument.Select();
}
         Function BtnSelectNextWord() {
objPlugIn.AuthenticView.Selection.SelectNext(1).Select(); }
         Function BtnSortDepartmentOnClick()
         {
            var objCursor = Null;
            var objTableStart = Null;
            var objBubble = Null;
            var strField1 = "";
            var strField1 = "";
            var nColIndex = 0;
            var nRows = 0;

            objCursor = objPlugIn.AuthenticView.Selection;
            If (objCursor.IsInDynamicTable())
            {
               // calculate current column index
               nColIndex = 0;
               While (True)
               {
                  try { objCursor.GotoPrevious(11); }
               catch (err) { break; }
               nColIndex++;
            }

            // GoTo begin of table
            objTableStart = objCursor.ExpandTo(9).CollapsToBegin().Clone();

            // count number of table rows
            nRows = 1;
            While (True)
```

```
            {
               try { objTableStart.GotoNext(10); }
            catch (err) { break; }
            nRows++;
            }

            // bubble sort through table
            For (var i = 0; i < nRows - 1; i++) {
               for(var j = 0; j < nRows-i-1; j++) {
                  objBubble = objCursor.ExpandTo(9).CollapsToBegin().Clone();
                  // Select correct column in jth table row
                  objBubble.GotoNext(6).Goto(10,j,2).Goto(11,nColIndex,2).ExpandTo(6);
                  strField1 = objBubble.Text;
                  strField2 =
objBubble.GotoNext(10).Goto(11,nColIndex,2).ExpandTo(6).Text;
                  if(strField1 > strField2) {
                     if(!objBubble.MoveRowUp()) {
                           alert('Table row move is not allowed!');
                           return;
                     }
                  }
               }
            }
         }
      </SCRIPT>
   </head>

   <body>
      <Object id="objPlugIn"
      <!-- CodeBase selects 32-bit CAB file (AuthenticBrowserEdition.CAB) -->
      <!-- or 64-bit Cab file (AuthenticBrowserEdition_x64.CAB) -->
         codeBase="
http://myCabfileLocation/AuthenticBrowserEdition.CAB#Version=12,2,0,0 "
       <!-- Class Id for 32-bit and 64-bit CAB files is the same -->
         classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780 "
         width="100%"
         height="80%"
         VIEWASTEXT>
         <PARAM NAME="EntryHelpersEnabled" VALUE="TRUE">
         <PARAM NAME="SaveButtonAutoEnable" VALUE="TRUE">
      </Object>
      <TABLE>
         <TR>
            <TD><Input Type="button" value="Goto Begin" id="B1" onclick="
BtnDocumentBegin()"></TD>
            <TD><Input Type="button" value="Goto End" name="B2" onclick="
BtnDocumentEnd()"></TD>
            <TD><Input Type="button" value="Whole Document" name="B3" onclick="
BtnWholeDocument()"></TD>
            <TD><Input Type="button" value="Select Next Word" name="B4" onclick="
BtnSelectNextWord()"></TD>
         </TR>
         <TR>
            <TD><Input Type="button" value="Sort Table by this Column" id="B6" onclick="
BtnSortDepartmentOnClick()"</TD>
         </TR>
      </TABLE>
      <TABLE id=SelTable border=1>
         <TR><TD id=SelTable_FirstTextPosition></TD><TD id=SelTable_LastTextPosition></
TD></TR>
         <TR><TD id=SelTable_FirstXMLData></TD><TD id=SelTable_FirstXMLDataOffset></TD
></TR>
         <TR><TD id=SelTable_LastXMLData></TD><TD id=SelTable_LastXMLDataOffset></TD></
TR>
         <TR><TD id=SelTable_Text></TD></TR>
      </TABLE>
   </body>

   <SCRIPT LANGUAGE=javascript For=objPlugIn EVENT=selectionchanged>
      var CurrentSelection =  Null;
      CurrentSelection = objPlugIn.AuthenticView.Selection;
      SelTable_FirstTextPosition.innerHTML = CurrentSelection.FirstTextPosition;
      SelTable_LastTextPosition.innerHTML = CurrentSelection.LastTextPosition;
      SelTable_FirstXMLData.innerHTML = CurrentSelection.FirstXMLData.Parent.Name;
      SelTable_FirstXMLDataOffset.innerHTML = CurrentSelection.FirstXMLDataOffset;
      SelTable_LastXMLData.innerHTML = CurrentSelection.LastXMLData.Parent.Name;
      SelTable_LastXMLDataOffset.innerHTML = CurrentSelection.LastXMLDataOffset;
   </SCRIPT>
```


© 2010 Altova GmbH

```
</html>
```

2.5 IE Beispiel 3: DB-Referenz

Von der unten gezeigten HTML-Seite wird ein auf einer Datenbank basierendes StyleVision Power Stylesheet (SPS) geladen. Alle für die Verbindung zur Datenbank benötigten Informationen sind im SPS in einem Connection String gespeichert. Die HTML-Datei darf keine Referenz auf eine Schema-Datei oder eine XML-Datei enthalten. Die Schema-Datei und die XML-Datei werden beim Herstellen der Datenbank-Verbindung automatisch anhand der Datenbank generiert. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code.
Im unten gezeigten Beispiel wird die Trusted Unicode-Version von Authentic Browser verwendet.

```
<html>
   <head>
       <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
       <title>Product Development Featurelist</title>
       <!-- Script for handling the ControlInitialized event -->
       <SCRIPT LANGUAGE="javascript" FOR=objPlugIn EVENT="ControlInitialized">
           serverstr = "dev37";
           objPlugIn.DesignDataLoadObject.URL = "http://"+ serverstr +
"/DBTest/DBSample.sps";
           objPlugIn.StartEditing();
       </SCRIPT>
   </head>
   <body>
       <OBJECT id="objPlugIn"
      CodeBase="http://yourserver/AuthenticBrowserEdition.CAB#Version=7,0,N,0"
         Classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780"  width="100%" height="100%
">
       </OBJECT>
   </body>

</html>
```

Nähere Informationen zu Connection Strings finden Sie unter "Voraussetzungen für DB-basierte SPS-Dateien".

© 2010 Altova GmbH

3 Firefox

Wenn die HTML-Seite für das Authentic Plug-in in Firefox geöffnet werden soll, muss sie die folgenden Elemente enthalten:

- ein HTML EMBED-Element, das (i) die DLL für das Authentic Plug-In vom Server auf den Client lädt und (ii) die Größe des Authentic-Fensters im Browser des Client definiert. Das EMBED-Element enthält den Pfad zum Authentic Browser Plug-in.
- eines oder mehrere Event Listeners zum Definieren der Subroutinen und zum Behandeln der Events. Event Listeners werden innerhalb von SCRIPT-Elementen erstellt.

Dieser Abschnitt ist in die folgenden Unterabschnitte gegliedert:

- Das EMBED-Element: Hier wird beschrieben, wie das HTML EMBED-Element in einer HTML-Seite für das Authentic Plug-in verwendet wird.
- Das SCRIPT-Element: Hier wird beschrieben, wie HTML SCRIPT Elemente in einer HTML-Seite für das Authentic Plug-in verwendet werden.
- Beispiele für komplette HTML-Seiten: IE Beispiel 1, einfach, IE Beispiel 2: Tabelle sortieren und IE Beispiel 3: Datenbank-Referenz.
-

Informationen zu einzelnen Objekten finden Sie im Abschnitt "Benutzerreferenz" unter der entsprechenden Object-Beschreibung.

Hinweis:

- Das Authentic Browser Plug-in wird für **Firefox 3.0 oder höher unterstützt.**
- Stellen Sie sicher, dass der MIME Type für XPI-Dateien im Browser-Dienst Ihres Servers zur Liste der MIME Types für die gewünschte Site hinzugefügt wurde.

3.1 Das EMBED-Element

Das EMBED-Element hat die folgenden Funktionen:

- Es gibt dem Authentic Browser Plug-In einen Namen (über das id-Attribut).
- Es wählt aus, welche Version des Authentic Browser Plug-In verwendet werden soll (über den Wert des type-Attributs).
- Es definiert über das PluginsPage-Attribut eine a XPI-Datei (diese ist die Authentic Browser Plug-in DLL). Der Wert des PluginsPage-Attributs ist ein Pfad zur XPI-Datei.
- Es definiert (über das height- und das width-Attribut) die Größe des Authentic -Ansichtsfensters im Browser des Client .

Hier sehen Sie ein Beispiel für ein HTML EMBED-Element. Es wählt die Trusted Unicode-Version aus (über den im type-Attribut definierten Wert) und legt für die Höhe und Breite des <%AUTH-VIEW%>-Ansichtsfensters im Browser des Client 100 % bzw. 60 % fest.

```
   <embed
       id="objPlugIn"
       type="application/x-authentic-scriptable-plugin"
       width="N00%"
       height="60%"
       PluginsPage="
http://your-server-including-path/AuthenticFirefoxPlugin_trusted.xpi"
       [Name of Parameter="Value of Parameter"]/>
```

id
Der Wert des id-Attributs wird als Name der Authentic Browser Plug-In Objekte verwendet, wenn diese in Skripts verwendet werden. So wird z.B. mit objPlugIn.SchemaLoadObject.URL das Objekt aufgerufen, das die Schema-Datei lädt.

type
Dieser Wert ist der MIME Type der benötigten Authentic Browser Version. Im oben aufgelisteten Code steht der Wert application/x-authentic-scriptable-plugin für die Trusted Version in englischer Sprache. Eine Liste der verfügbaren Versionen und MIME Types finden Sie im Abschnitt Authentic Browser Versionen.

width, height
Mit diesen Attributen wird die Größe des im Browser-Fenster anzulegenden Authentic -Ansichtsfensters definiert.

PluginsPage
Der Wert dieses Attributs gibt den Pfad zur Authentic Browser XPI-Datei auf Ihrem Server an. Stellen Sie sicher, dass Sie in der URL für die XPI-Datei den richtigen Pfad verwenden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Pfaden im Code.

Parameter
Ein Parameter (*siehe Liste weiter unten*) kann auf die folgenden Arten verwendet werden:

- Durch Angabe des Parameternamens und seines Werts als Attributwertpaar des EMBED -Elements. So können Sie z.B. den Parameter ToolbarsEnabled mit einem Wert true definieren, indem Sie das Attributwertpaar ToolbarsEnabled="true" zum EMBED Element hinzufügen. Siehe Liste oben.
- Ein PARAM-Element kann als Child des OBJECT-Elements definiert werden, wie im Beispiel unten:

```
<object id="objPlugIn"
```


© 2010 Altova GmbH

```
        type="application/x-authentic-scriptable-plugin"
        width="N00%"
        height="60%" >
    <param name="ToolbarsEnabled"
        value="true"/>
</object>
```

Beachten Sie, dass die Definition von Parametern auf diese Art, d.h. als Child des `OBJECT`-Elements auch einen Nachteil hat. Da Firefox das `PLUGINSPAGE` Attribut des `OBJECT`-Elements nicht akzeptiert, verfügt Firefox über keine Referenz auf die Authentic Browser XPI-Datei auf dem Server und kann daher die Installation des Authentic Browser Plug-in auf dem Client nicht starten. Daher eignet sich diese Art der Parameterdefinition nur auf Clients, auf denen das Plug-in bereits installiert ist.

Es können beliebig viele der folgenden Parameter verwendet werden.

LicServer
Der Name des Servers, für den die Authentic Browser Enterprise Edition-Lizenz gültig ist. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

LicKey
Der Lizenzschlüssel, der für die Authentic Browser Enterprise Edition benötigt wird. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

LicCompany
Der Name des Unternehmens, für das die Verwendung der Authentic Browser Enterprise Edition lizenziert wurde. (Für die Authentic Browser Community Edition wird kein Lizenzschlüssel benötigt.)

XMLDataURL
Eine absolute URL, die den Speicherort der zu editierenden XML-Datei angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

XMLDataSaveURL
Eine absolute URL, die den Ordner angibt, in dem die XML-Datei gespeichert werden soll. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

SPSDataURL
Eine absolute URL, die den Pfad zum StyleVision Power Stylesheet (`.sps`-Datei) angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

SchemaDataURL
Eine absolute URL, die den Pfad zur verknüpften Schema-Datei angibt. Für Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad angeben.

TextStateBmpURL
Der Ordner, in dem Bitmap-Grafiken für Textstatus-Symbole gespeichert werden sollen.

TextStateToolbarLine
Die Zeile der Symbolleiste, in der Textstatus-Symbole platziert werden sollen. Die Standardeinstellung ist 1.

AutoHideUnusedCommandGroups
Gibt an, ob Gruppen nicht verwendeter Symbolleistenbefehle ausgeblendet werden sollen. Die Standardeinstellung ist True.

ToolbarsEnabled
Gibt die allgemeine Unterstützung für Symbolleisten an. Die Standardeinstellung ist True.

ToolbarTooltipsEnabled
Gibt an, ob Tooltipps aktiviert oder deaktiviert sind.

HideSaveButton
Bei Einstellung "True" wird die standardmäßig eingeblendete Schaltfläche "Speichern" aus der Authentic-Symbolleiste entfernt.

BaseURL
Gibt die Basis-URL für relative Pfade an.

SaveButtonUsePOST
Bei Einstellung "True" wird beim Speichern des Dokuments der Befehl HTTP POST verwendet, anstelle von PUT.

EntryHelpersEnabled
Bei Einstellung "True" werden die Authentic-Eingabehilfen eingeblendet.

EntryHelperSize
Breite der Eingabehilfefenster in Pixeln.

EntryHelperAlignment
Definiert die Position der Eingabehilfen im Dokumentenfenster.

- `0` = Symbolleiste am oberen Rand des Dokuments anzeigen
- `N` = Symbolleiste am linken Rand des Dokuments anzeigen
- `2` = Symbolleiste am unteren Rand des Dokuments anzeigen
- `3` = Symbolleiste am rechten Rand des Dokuments anzeigen

EntryHelperWindows
Gibt an, welche der Eingabehilfenfenster eingeblendet werden sollen.

- `N` = Elemente
- `2` = Attribute
- `4` = Entities

Zulässig ist jede Kombination (bit-check)

SaveButtonAutoEnable
Siehe Authentic.SaveButtonAutoEnable

LoaderSettingsFileURL
Gibt die URL der `LoaderSettingsFile` für die Paketverwaltung an.

Die folgenden Parameter werden nicht mehr verwendet:
Ab Version 6.0.3.1 von Authentic Browser sind keine Lizenzen mehr erforderlich.

LicServer
Der Name des Servers, auf dem sich der Lizenzkey befindet (falls der Lizenzkey über eine Datei zur Verfügung gestellt wird).

LicFile
Der Name der Datei, in der der Lizenzkey gespeichert ist.

LicKey
Der Lizenzkey zur Überprüfung der Verwendung der Authentic Browser Edition.

LicCompany

© 2010 Altova GmbH

Der Name des Unternehmens (zur Überprüfung der Verwendung der Authentic Browser Edition).

LicUser
Der Name des Benutzers (zur Überprüfung der Verwendung der Authentic Browser Edition).

3.2 Hinzufügen von Event Listener-Funktionen

Das folgende SCRIPT Element definiert eine Event Listener-Funktion und registriert sie beim Plug-in-Objekt. Die Event Listener-Funktion wird innerhalb des Plug-in ausgelöst.

```
<SCRIPT LANGUAGE="javascript">
var selCount = 0;
function OnSelectionChanged()
{
        selCount = selCount + 1;
        selectionCounter.value = "SelectionCount = " + selCount;
}
var objPlugIn = document.getElementById('objPlugIn');
objPlugIn.addEventListener("selectionchanged", OnSelectionChanged, false)
</SCRIPT>
```

Eine Liste von Events finden Sie außerdem unter Events: Referenz

Sprache
Das Authentic Browser Plug-in wurde mit JavaScript und VBScript getestet.

Event listener-Funktionen
Informationen zu Event Listener-Funktionen finden Sie in der entsprechenden W3C Recommendation.

Eine Liste von Ereignissen finden Sie auch unter Ereignisse: Referenz.

Authentic Browser Objektmodell
Die Methoden, Eigenschaften und Objekte im Authentic Browser Plug-in sind im Abschnitt Benutzerreferenz dieser Dokumentation näher beschrieben.

© 2010 Altova GmbH

3.3 Firefox Beispiel 1: Einfach

Der unten gezeigte HTML-Code generiert eine Seite, die die folgenden Funktionen hat:

- Sie installiert die Trusted Unicode-Version von Authentic Browser für Firefox am Client, falls dies nicht bereits geschehen ist.
- Das Authentic Browser-Fenster innerhalb dieser Seite hat eine Breite, die 100 % von der des Browser-Fensters beträgt und 60 % seiner Höhe.
- Unterhalb des Authentic Browser-Fensters befindet sich eine Reihe mit vier Schaltflächen
- Die Authentic-Ansicht von `OrgChart.xml` ist geladen.
- Die Schaltflächen **Suchen** und **Ersetzen** dienen zum Aufrufen des Such- bzw. Ersetzungsdialogfelds.
- Die Schaltfläche **Speichern** dient zum Speichern der Änderungen in einer Datei mit dem Namen `SaveFile.xml`, die im Root-Verzeichnis des Servers abgelegt ist.
- Die Schaltfläche **Eigenschaft testen** dient zum Testen einer einfachen Eigenschaft.

Wenn diese HTML-Seite auf dem Client geöffnet wird, kann der Benutzer die XML-Datei `OrgChart.xml` bearbeiten und die bearbeitete Datei als `SaveFile.xml` speichern.

Anhand dieser einfachen HTML-Seite können Sie testen, ob Authentic Browser korrekt funktioniert. Stellen Sie dabei sicher, dass Sie die richtige IP-Adresse und den richtigen Pfad zu den entsprechenden Dateineen in den URLs der `XPI`-Datei, der `xsd`-, `xml`- und der `sps`-Datei und aller anderen auf dem Server gespeicherten Ressourcen verwenden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code. Sie können dieses Beispiel erweitern und ändern, um komplexere Lösungen mit Hilfe von Authentic Browser zu erstellen.

```
<html>
<head>
   <meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
   <title>Minimal XMLSpyDocEditPlugIn page</title>

</head>

<!-- to disable the fast-back cache in Firefox, define an unload handler -->
<BODY id="bodyId" onunload="Unload()">
   <!-- Embed element has id with value that must be used -->
   <!-- as name of Authentic Browser Plug-in objects -->
   <!-- type selects the Trusted Unicode version -->
   <embed
        id="objPlugIn"
        type="application/x-authentic-scriptable-plugin"
        width="100%"
        height="60%"
        PLUGINSPAGE="
http://your-server-including-path/AuthenticFirefoxPlugin_trusted.xpi "
        LicServer="DevAuthBrowTest"
        LicCompany="Altova"
        LicKey="XXXXXXXX" />
   <!-- Script with subroutines -->
   <SCRIPT LANGUAGE="javascript">
        var objPlugIn = document.getElementById('objPlugIn');
        function OnClickFind()
        {
               objPlugIn.FindDialog();
        }

        function OnClickReplace()
        {
               objPlugIn.ReplaceDialog();
        }

        function BtnOnSave()
        {
               objPlugIn.XMLDataSaveUrl = "http://your-server/Authentic/SaveFile.xml"
```

```
            objPlugIn.Save()
        }

        function BtnOnTestProp()
        {
            alert ( objPlugIn.IsRowInsertEnabled );
        }
        function Unload()
        {
        }
        function InitAuthenticPluginPage( )
        {
            var serverstr='your-server/';
            var basedir='Authentic/';
            objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
            objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
            objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
            objPlugIn.StartEditing();
        }
        // event subscription if running on Firefox
        objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage,
false);
    </SCRIPT>
    <p>
      <input type="button" value="Find" name="B4" onclick="OnClickFind()">
      <input type="button" value="Replace" name="B5" onclick="OnClickReplace()">
      <input type="button" value="Save" name="B6" onclick="BtnOnSave()">
      <input type="button" value="Test property" name="B7" onclick="BtnOnTestProp()">
    </p>
</body>
</html>
```

Anmerkung: Das obige Script enthält Lizenzinformationen zum Aktivieren der Authentic Browser Enterprise Edition. Wenn die drei Parameter `LicServer`, `LicCompany` und `LicKey` fehlen, so stehen nur die Funktionalitäten der Authentic Browser Community Edition zur Verfügung.

© 2010 Altova GmbH

3.4 Firefox Beispiel 2: Tabelle sortieren

Dies ist eine HTML-Beispieldatei mit einem eingebetteten JavaScript. Für das Beispiel muss das Authentic Browser Plug-In (XPI-Datei) auf Ihrem Computer installiert werden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code.

Der Code zeigt an:

- wie das Browser Plug-In aufgerufen werden soll. Passen Sie den Code bitte an und geben Sie den Pfad zu Ihrer XPI-Datei und dem Class Identifier (MIME Type) Ihrer Browser Plug-In-Version an (trusted oder untrusted).
- wie eine Datei in das Browser Plug-In geladen werden soll. Passen Sie bitte den Code an und geben Sie den Pfad zu Ihrem Beispieldokument an.
- wie Schaltflächen für einfache Cursor-Positionierungen implementiert werden.
- wie komplexere Befehle wie z.B. die Sortierung von Tabellen implementiert werden.
- wie das `SelectionChanged`-Ereignis verwendet wird.

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>Test Page For Authentic Browser Plug-in</title>

</head>

<!-- to disable the fast-back cache in Firefox, define an unload handler -->
<BODY id="bodyId" onunload="Unload()">
   <embed
        id="objPlugIn"
        type="application/x-authentic-scriptable-plugin "
        width="100%"
        height="60%"
        PLUGINSPAGE="
http://your-server-including-path/AuthenticFirefoxPlugin_trusted.xpi "
        EntryHelpersEnabled="TRUE"
        SaveButtonAutoEnable="TRUE" >
   </embed>
<TABLE>
<SCRIPT LANGUAGE="javascript">
var objCurrentRange = null;
var objPlugIn = document.getElementById('objPlugIn');

function BtnDocumentBegin() { objPlugIn.AuthenticView.DocumentBegin.Select(); }
function BtnDocumentEnd() { objPlugIn.AuthenticView.DocumentEnd.Select(); }
function BtnWholeDocument() { objPlugIn.AuthenticView.WholeDocument.Select(); }
function BtnSelectNextWord() { objPlugIn.AuthenticView.Selection.SelectNext(1).Select();
}

function BtnSortDepartmentOnClick()
{
        var objCursor = null;
        var objTableStart = null;
        var objBubble = null;
        var strField1 = "";
        var strField1 = "";
        var nColIndex = 0;
        var nRows = 0;

        objCursor = objPlugIn.AuthenticView.Selection;
        if (objCursor.IsInDynamicTable())
        {
                // calculate current column index
                nColIndex = 0;
                bContinue = true;
                while ( bContinue )
                {
                        try { objCursor.GotoPrevious(11); }
                        catch (err) { bContinue = false; nColIndex--; }
                        nColIndex++;
                }
```

```
                // GoTo begin of table
                objTableStart = objCursor.ExpandTo(9).CollapsToBegin().Clone();

                // count number of table rows
                nRows = 1;
                bContinue = true;
                while ( bContinue )
                {
                        try { objTableStart.GotoNext(10); }
                        catch (err) { bContinue = false; }
                        nRows++;
                }

                // bubble sort through table
                for ( i = 0; i < nRows - 1; i++) {
                        for( j = 0; j < nRows-i-1; j++) {
                                objBubble =
objCursor.ExpandTo(9).CollapsToBegin().Clone();
                                // Select correct column in jth table row

objBubble.GotoNext(6).Goto(10,j,2).Goto(11,nColIndex,2).ExpandTo(6);
                 strField1 = objBubble.Text;
                                try
                                {
                                        strField2 =
objBubble.GotoNext(10).Goto(11,nColIndex,2).ExpandTo(6).Text;
                                }
                                catch ( err ) { continue; };
                  if(strField1 > strField2) {
                                        if(!objBubble.MoveRowUp()) {
                                                alert('Table row move is not allowed!');
                                                return;
                                        }
                                }
                        }
                }
        }
}

function InitAuthenticPluginPage( )
{
        var serverstr='your-server/';
        var basedir='Authentic/';
        objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
        objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
        objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
        objPlugIn.StartEditing();
}

function Unload()
{
}

function OnSelectionChanged()
{
        var CurrentSelection = null;
        CurrentSelection = objPlugIn.AuthenticView.Selection;
        SelTable_FirstTextPosition.innerHTML = CurrentSelection.FirstTextPosition;
        SelTable_LastTextPosition.innerHTML = CurrentSelection.LastTextPosition;
        SelTable_FirstXMLData.innerHTML = CurrentSelection.FirstXMLData.Parent.Name;
        SelTable_FirstXMLDataOffset.innerHTML = CurrentSelection.FirstXMLDataOffset;
        SelTable_LastXMLData.innerHTML = CurrentSelection.LastXMLData.Parent.Name;
        SelTable_LastXMLDataOffset.innerHTML = CurrentSelection.LastXMLDataOffset;
}

objPlugIn.addEventListener("selectionchanged", OnSelectionChanged, false)
// event subscription if running on Firefox
objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage, false);
</SCRIPT>
<TR>
<TD><Input Type="button" value="Goto Begin" id="B1" onclick="BtnDocumentBegin()"></TD>
<TD><Input Type="button" value="Goto End" name="B2" onclick="BtnDocumentEnd()"></TD>
<TD><Input Type="button" value="Whole Document" name="B3" onclick="BtnWholeDocument()
"></TD>
<TD><Input Type="button" value="Select Next Word" name="B4" onclick="BtnSelectNextWord()
```

 © 2010 Altova GmbH

```
"></TD>
</TR>
<TR>
<TD><Input Type="button" value="Sort Table by this Column" id="B6" onclick="
BtnSortDepartmentOnClick()"</TD>
</TR>
</TABLE>
<TABLE id=SelTable border=1>
<TR><TD id=SelTable_FirstTextPosition></TD><TD id=SelTable_LastTextPosition></TD></TR>
<TR><TD id=SelTable_FirstXMLData></TD><TD id=SelTable_FirstXMLDataOffset></TD></TR>
<TR><TD id=SelTable_LastXMLData></TD><TD id=SelTable_LastXMLDataOffset></TD></TR>
<TR><TD id=SelTable_Text></TD></TR>
</TABLE>
</body>
</html>
```

4 Browser-unabhängig

In manchen Projekten ist es nicht immer bekannt, welcher Browser (Internet Explorer oder Firefox) auf dem Client verwendet wird. In diesen Fällen können auf dem Server Authentic Browser Versionen sowohl für Internet Explorer als auch Firefox gespeichert werden (d.h. sowohl die CAB-Datei als auch die XPI-Datei kann auf dem Server gespeichert werden). Sie können in die HTML-Seite ein Script einfügen, mit dem ermittelt wird, mit welchem Browser die HTML-Seite geöffnet wurde, sodass die richtige Authentic Browser Plug-in geladen werden kann.

Wenn auf dem Client Internet Explorer verwendet wird, so muss die richtige . `CAB` Datei (für den 32-Bit oder den 64-Bit Internet Explorer) ausgewählt werden, um vom Server heruntergeladen zu werden. Mit Hilfe eines Skripts kann ermittelt werden, welche X-Bit Version von Internet Explorer verwendet wird, damit die richtige . `CAB` Datei vom Server heruntergeladen wird.

Der Abschnitt enthält eine Beispieldatei, in der der Browser ermittelt wird, die korrekte Authentic Browser Version geladen wird und einige Funktionen ausgeführt werden.

Informationen zu einzelnen Objekten finden Sie im Abschnitt "Benutzerreferenz" unter der entsprechenden Object-Beschreibung.

Hinweis:

- Das Authentic Browser Plug-in wird für **Internet Explorer 5.5 oder höher** und **Firefox 3.0 oder höher unterstützt.**
- Stellen Sie sicher, dass der MIME Type für XPI-Dateien im Browser-Dienst Ihres Servers zur Liste der MIME Types für die gewünschte Site hinzugefügt wurde.

© 2010 Altova GmbH

4.1 Abrufen des richtigen Plug-in

Falls nicht bekannt ist, welcher Browser auf dem Client-Rechner verwendet wird, kann ein Script im HTML-Code zuerst ermitteln, welcher Browser verwendet wird und daraufhin das entsprechende Code-Fragment auswählen.

Verwendung des korrekten Plug-in zum Aufrufen einer JavaScript-Funktion
Das folgende Script muss in den Head-Abschnitt eingefügt werden. Es definiert eine JavaScript-Funktion, die aufgerufen wird, wenn das <body> Element des HTML-Codes geladen wird. Das <body> Element würde das Script enthalten, die die Art des Browsers ermittelt.

```
<SCRIPT type="text/javascript" LANGUAGE="javascript" >
function InitAuthenticPluginPage( )
{
	var objPlugIn = document.getElementById('objPlugIn');
	var serverstr='yourserver/';
	var basedir='Authentic/';
	objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
	objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
	objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.sps';
	objPlugIn.StartEditing();
}
</SCRIPT>
```

Ermitteln des vom Client verwendeten Browser-Typs
Das folgende Script ermittelt, mit welchem Browser die Seite geladen wird. Es erstellt je nachdem, welcher Browser ermittelt wurde, das Plug-in-Objekt (für Internet Explorer oder Firefox)

```
<script language="JavaScript">
function isFirefoxOnWindows()
{
	return ((navigator.userAgent.indexOf('Firef') != -1) &&
(navigator.userAgent.indexOf('Win') != -1));
}

function isIEOnWindows()
{
	return ((navigator.userAgent.indexOf('MSIE') != -1) &&
(navigator.userAgent.indexOf('Win') != -1))
}

function createObject( codebase, clsid)
{
	if ( isFirefoxOnWindows() )
	{
		document.write ( '<embed ' +
                                                    'id="objPlugIn" ' +
                                                    'type="
application/x-authentic-scriptable-plugin" ' +
                                                    'width="100%" ' +
                                                    'height="60%" PLUGINSPAGE="
http://yourserver/Authentic/AuthenticFirefoxPlugin_trusted.xpi ">' +
                                            '</embed>' );
	}
	else if ( isIEOnWindows() )
	{
		document.write ( '<OBJECT ' +
                                                'id="objPlugIn" ' +
                                                'codebase="
http://yourserver/Authentic/AuthenticBrowserEdition.CAB " ' +
                                                'Classid="
clsid:B4628728-E3F0-44a2-BEC8-F838555AE780 " ' +
                                                'width="100%" ' +
                                                'height="60%" ' +
                                                '>' +
```

```
                                                    '</OBJECT>');
        }
}

createObject();
</script>
```


© 2010 Altova GmbH

4.2 Browser-unabhängiges Beispiel

Der unten gezeigte HTML-Code generiert eine Seite, die die folgenden Funktionen hat:

- Sie überprüft, welcher Browser auf dem Client installiert ist (Internet Explorer oder Firefox) und installiert die Authentic Browser Version für den entsprechenden Browser.
- Wenn als Browser Internet Explorer installiert ist, so wird überprüft, ob es sich um ein 32-Bit oder ein 64-Bit-System handelt. Anschließend wird die richtige .CAB-Datei (für den 32-Bit oder den 64-Bit Internet Explorer) ausgewählt.
- Das Authentic Browser-Fenster in der Seite hat eine Breite, die 100 % von der des Browser-Fensters beträgt und 60 % seiner Höhe.
- Unterhalb des Authentic Browser-Fensters befindet sich eine Reihe mit fünf Schaltflächen
- Die Schaltfläche **Bearbeitung starten** dient zum Laden der Authentic-Ansicht von `OrgChart.xml`, welche sich im Root-Verzeichnis Ihres Servers befindet.
- Die Schaltflächen **Suchen** und **Ersetzen** dienen zum Aufrufen des Such- bzw. Ersetzungsdialogfelds.
- Die Schaltfläche **Speichern** dient zum Speichern der Änderungen in einer Datei mit dem Namen `SaveFile_OrgChart.xml`, die im Root-Verzeichnis des Servers abgelegt ist.
- Die Schaltfläche **Eigenschaft testen** dient zum Testen einer einfachen Eigenschaft.

Wenn diese HTML-Seite auf dem Client geöffnet wird, kann der Benutzer die XML-Datei `OrgChart.xml` bearbeiten und die bearbeitete Datei als `SaveFile_OrgChart.xml` speichern.

Anhand dieser einfachen HTML-Seite können Sie testen, ob Authentic Browser korrekt funktioniert. Stellen Sie dabei sicher, dass Sie die richtige IP-Adresse und den richtigen Pfad zu den entsprechenden Dateineen in den URLs der `XPI`-Datei, der `xsd`-, `xml`- und der `sps`-Datei und aller anderen auf dem Server gespeicherten Ressourcen verwenden. Beachten Sie, dass die Groß- und Kleinschreibung auf einigen Servern eine Rolle spielt. Wenn eine Datei daher nicht gefunden wird, überprüfen Sie die Groß- und Kleinschreibung von Dateinamen und Befehlen im Code. Sie können dieses Beispiel erweitern und ändern, um komplexere Lösungen mit Hilfe von Authentic Browser zu erstellen.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
   <head>
      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
      <title>Orgchart.sps Scriptable Plug-in Test - browser independent</title>
      <script type="text/javascript">
         <!--
         function BtnOnSave() { objPlugIn.Save();}

         function InitAuthenticPluginPage( )
         {

          var schema= document.getElementById('xsd');
          var instance=document.getElementById('xml');
          var design=document.getElementById('sps');
          objPlugIn.XMLDataLoadObject.URL =instance.innerHTML;
          objPlugIn.DesignDataLoadObject.URL = design.innerHTML;
          objPlugIn.SchemaLoadObject.URL= schema.innerHTML;
         // alert(schema.innerHTML+"  "+instance.innerHTML+" " +design.innerHTML);

         /*
          var serverstr='your-server/';
          var basedir='Authentic/';
          objPlugIn.SchemaLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xsd';
          objPlugIn.XMLDataLoadObject.URL = 'http://' + serverstr + basedir +
'OrgChart.xml' ;
          objPlugIn.DesignDataLoadObject.URL = 'http://' + serverstr + basedir +
```

```
'OrgChart.sps';
         */

          objPlugIn.StartEditing();


         }

         function Unload()
         {
         }
         //-->
      </script>
   <style type="text/css">@page { margin-left:0.60in; margin-right:0.60in;
margin-top:0.79in; margin-bottom:0.79in } @media     screen { br.altova-page-break {
display: none; } } @media print { br.altova-page-break { page-break-before: always; } }
   </style>
   </head>

   <body id="bodyId" onunload="Unload()">
      <table border="1">
         <tbody>
            <tr><th><span>DesignLoadURL</span></th><td id="sps">
http://your-server/Authentic/Orgchart.sps </td></tr>
            <tr><th><span>SchemaLoadURL</span></th><td id="xsd">
http://your-server/Authentic/Orgchart.xsd </td></tr>
            <tr><th><span>XMLDataLoadURL</span></th><td id="xml">
http://your-server/Authentic/Orgchart.xml </td></tr>
            <tr><th><span>XMLDataSaveURL</span></th><td id="xmlsave">
http://your-server/Authentic/SaveFile_OrgChart.xml </td></tr>
         </tbody>
      </table>
      <center><h3><span>Authentic Platformindependent Plug-in Enterprise Edition</span
></h3></center>
      <span> </span>
      <center>
      <script language="JavaScript">
         // return true if the page loads in Firefox
         function isFirefoxOnWindows()
         {
          return ((navigator.userAgent.indexOf('Firef') != -1) &&
(navigator.userAgent.indexOf('Win') != -1));
         }

         // return true if the page loads in Internet Explorer
         function isIEOnWindows()
         {
          return ((navigator.userAgent.indexOf('MSIE') != -1) &&
(navigator.userAgent.indexOf('Win') != -1))
         }

         //return true if Browser is 64bit
         function is64bitBrowser()
         {
          return ((navigator.userAgent.indexOf('Win64') != -1)&&
(navigator.userAgent.indexOf('x64') != -1))
         }

         //return Codebase for 32 bit  or 64 bit
         function getCodeBase()
         {
           if ( is64bitBrowser() ){

return('CodeBase="http://your-server/AuthenticBrowserEdition_x64.CAB#Version=12,2,0,0"
')
           }
           else {

return('CodeBase="http://your-server/AuthenticBrowserEdition.CAB#Version=12,2,0,0" ')
           }
         }

         // Create the plugin object instance, according to the browser loading the
page
         // -Firefox uses EMBED tag for embeding plugins and supports PLUGINSPAGE
         // attribute to redirect to an installation file if the plugin is not
         // currently installed;
         // -IE uses <OBJECT> tag for embeding plugins and supports CODEBASE attribute
         // to indicate a .cab file for the installation if the plugin is not
```

 © 2010 Altova GmbH

```
            // currently installed

            function createObject( codebase, clsid)
            {
             if ( isFirefoxOnWindows() )
             {
             document.write ( '<embed ' +
             'id="objPlugIn" ' +
             'type="application/x-authentic-scriptable-plugin" ' +
             'width="100%" ' +
             'height="60%" ' + PLUGINSPAGE="
http://your-server/Authentic/AuthenticFirefoxPlugin_trusted.xpi " ' +
             'SaveButtonAutoEnable="true" ' +
             'EntryHelpersEnabled="true" ' +
             'LicServer="your-server" ' +
             'LicCompany="Altova" ' +
             'LicKey="XXXXXXXXXX" ' +
             'XMLDataSaveUrl="http://your-server/Authentic/SaveFile_OrgChart.xml "> ' +
             '</embed>' );
             }
              else if ( isIEOnWindows() )
             {
             document.write ( '<OBJECT ' +
               'id="objPlugIn" ' +
               getCodeBase() +
               'Classid="clsid:B4628728-E3F0-44a2-BEC8-F838555AE780 " ' +
               'width="100%" ' +
               'height="60%" ' +
               '>' +
               '<PARAM NAME="XMLDataSaveUrl" VALUE="
http://your-server/Authentic/SaveFile_OrgChart.xml "> ' +
               '<PARAM NAME="EntryHelpersEnabled" VALUE="TRUE"> ' +
               '<PARAM NAME="SaveButtonAutoEnable" VALUE="TRUE"> ' +
               '<PARAM NAME="LicServer" VALUE="your-server"> ' +
               '<PARAM NAME="LicCompany" VALUE="Altova"> ' +
               '<PARAM NAME="LicKey" VALUE="XXXXXXXXXX"> ' +
               '</OBJECT>');
            }
            }

            createObject();
            // after running createObject the plugin object exists. Initialize the
javascript variable to be used in the scripts
            var objPlugIn = document.getElementById('objPlugIn');
      </script>

            <br><br>
            <button onclick="objPlugIn.StartEditing()"><span>Start Editing</span></button>
            <button onclick="objPlugIn.FindDialog()"><span>Find</span></button>
            <button onclick="objPlugIn.ReplaceDialog();"><span>Replace</span></button>
            <button onclick="BtnOnSave()"><span>Save</span></button>
            <button onclick="alert ( objPlugIn.IsRowInsertEnabled );"><span>Test</span><br
></button>
            </center>

      <script language="javascript">
            // event subscription if running on Firefox
             if ( isFirefoxOnWindows() )
             {
             objPlugIn.addEventListener("ControlInitialized", InitAuthenticPluginPage,
false);
             }
      </script>

      <script event="ControlInitialized" for="objPlugIn" language="javascript">
            // event subscription if running on Internet Explorer
             if ( isIEOnWindows() )
             {
             InitAuthenticPluginPage();
             //if ( isIEx64OnWindows() ) alert("IE x64");
             }
      </script>

   </body>
</html>
```

Anmerkung: Das oben gezeigte Skript enthält Lizenzinformationen zum Aktivieren der

Authentic Browser Enterprise Edition. Wenn die drei Parameter `LicServer`, `LicCompany` und `LicKey` nicht vorhanden sind, so werden die Funktionalitäten auf die der Authentic Browser Community Edition beschränkt.

© 2010 Altova GmbH

Kapitel 7

Benutzerreferenz

Benutzerreferenz

In diesem Abschnitt sind die Mechanismen, Objekte und Enumerationen von Authentic Browser aufgelistet und beschrieben.

- Authentic Browser Mechanismen
- Authentic Browser Objekte
- Authentic Browser Enumerationen

© 2010 Altova GmbH

1 Mechanismen

Dieser Abschnitt enthält eine Beschreibung einiger der am häufigsten verwendeten Authentic Browser-Mechanismen.

1.1 Ereignisse: Connection Point (IE-spezifisch)

In Authentic Browser steht eine Reihe von Connection Point-Ereignissen zur Verfügung (siehe auch Ereignisse: Referenz), für die Sie Ereignishandler in den SCRIPT-Blöcken auf Ihrer HTML-Seite bereitstellen können.

Anmerkung: Die Beschreibung in diesem Abschnitt bezieht sich **nur auf Internet Explorer**.

In den folgenden Beispielen sehen Sie `SCRIPT` Blöcke für die Ereignisse "ControlInitialized" und "SelectionChanged":

`ControlInitialized`
Dieses Connection Point-Ereignis wird sofort nach Erstellung und Initialisierung des Controls ausgelöst. Ein zusätzliches Initialisierungsskript für das Control kann innerhalb der `ControlInitialized`-Ereignisbehandlungsroutine abgearbeitet werden.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=controlinitialized>
      // add your code here
</SCRIPT>
```

`SelectionChanged`
Das `SelectionChanged`-Ereignis wird jedes Mal ausgelöst, wenn die aktuelle Auswahl in der Ansicht geändert wird. Verwenden Sie einen `SCRIPT`-Block, um den jeweiligen Code für das Ereignis auszuführen.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=selectionchanged>
      // add your code here
</SCRIPT>
```

Bitte beachten Sie, dass das Authentic.event-Objekt nicht befüllt wird, wenn dieses Ereignis vorkommt. Wenn in Ihrem Skript Ereignishandler registriert sind, enthalten die Eigenschaften des Authentic.event-Objekts Werte aus dem letzten Ereignis, das aufgetreten ist.

Das Authentic.CurrentSelection-Objekt enthält nun gültige Informationen.

Laden von SPS-, XSD- und XML-Dateien ohne Zutun des Benutzers
Wenn die `.sps-`, `.xsd`- und die `.xml`-Datei beim Laden der HTML-Seite ohne Zutun des Benutzers geladen werden sollen, ist die beste Methode, einen Ereignishandler zu erzeugen, der das `ControlInitialized`-Event verarbeitet. Alternativ dazu können Sie auch ein PropertyBag verwenden (siehe zweites Beispiel):

Empfohlene Methode:

```
<SCRIPT LANGUAGE="javascript" FOR=objPlugIn
EVENT="ControlInitialized">
objPlugIn.SchemaLoadObject.URL = "http://yourserver/OrgChart.xsd"
objPlugIn.XMLDataLoadObject.URL = "http://yourserver/OrgChart.xml"
objPlugIn.DesignDataLoadObject.URL = "http://yourserver/OrgChart.sps"
objPlugIn.StartEditing()
</SCRIPT>
```

oder

```
<OBJECT id=objPlugIn style="WIDTH:600px; HEIGHT:500px"
codeBase="http://yourserver/cabfiles/AuthenticBrowserEdition.CAB#Versi
on=7,0,N,0"
classid=clsid:B4628728-E3F0-44a2-BEC8-F838555AE780>
```

 © 2010 Altova GmbH

```
<PARAM NAME="XMLDataURL" VALUE="http://yourserver/OrgChart.xml">
<PARAM NAME="SPSDataURL" VALUE="http://yourserver/OrgChart.sps">
<PARAM NAME="SchemaDataURL" VALUE="http://yourserver/OrgChart.xsd">
</OBJECT>
```

Es ist nicht ratsam, diese Dateien in einen Ereignishandler zu laden, der das "onload"-Ereignis des "body"-Elements abarbeitet, da das Authentic Browser Plug-In-Control möglicherweise erst initialisiert wird, nachdem das "onload"-Ereignis ausgelöst wurde. In diesem Fall stehen die Methoden und Eigenschaften des Plug-In nicht zur Verfügung und die Dateien werden nicht geladen.

Nicht empfohlen:

```
<SCRIPT LANGUAGE="javascript">
 function load () {

objPlugIn.SchemaLoadObject.URL = "http://yourserver/OrgChart.xsd"
objPlugIn.XMLDataLoadObject.URL = "http://yourserver/OrgChart.xml"
objPlugIn.DesignDataLoadObject.URL = "http://yourserver/OrgChart.sps"
objPlugIn.StartEditing()
}
</SCRIPT>

<body onload = "load files">
```

1.2 Ereignisse: Hinzufügen von Event Listener-Funktionen (Firefox-spezifisch)

In Authentic Browser steht eine Reihe von Connection Point-Ereignissen zur Verfügung (siehe auch Ereignisse: Referenz), für die Sie mit Hilfe der Methode `addEventListener` Ereignishandler in den SCRIPT-Blöcken auf Ihrer HTML-Seite bereitstellen können.

Anmerkung: Die Beschreibung in diesem Abschnitt **gilt nur für Firefox**.

Hier sehen Sie ein Beispiel, wie man für das SelectionChanged-Ereignis in Firefox einen Ereignis-Handler bereit stellt.

`SelectionChanged`

Das `SelectionChanged` Ereignis wird jedes Mal, wenn sich die aktuelle Auswahl in der Ansicht ändert, ausgelöst. Verwenden Sie einen `SCRIPT` Block, um Ihren ereignisspezifischen Code auszuführen.

```
<SCRIPT LANGUAGE="javascript">
var selCount = 0;
function OnSelectionChanged()
{
    selCount = selCount + N;
    selectionCounter.value = "SelectionCount = " + selCount;
}
var objPlugIn = document.getElementById('objPlugIn');
objPlugIn.addEventListener("selectionchanged", OnSelectionChanged, false)
</SCRIPT>
```

Beachten Sie bitte, dass das Authentic.event Objekt nicht befüllt wird, wenn dieses Ereignis eintrifft. Wenn in Ihrem Script Event Handler registriert sind, enthalten die Eigenschaften des Authentic.event Objekts Werte aus dem letzten Ereignis, das aufgetreten ist.

Das Authentic.CurrentSelection Objekt enthält nun gültige Informationen.

© 2010 Altova GmbH

1.3 Ereignisse: Symbolleisten-Schaltfläche

Jede Symbolleisten-Schaltfläche hat ein Standardverhalten, das unter Umständen geändert werden muss. Mit Hilfe des `AuthenticCommand`-Ereignisses können Sie zusätzliche Aufgaben hinzufügen oder das Standardverhalten eines solchen Schaltflächenbefehls völlig neu definieren. In Skripts kann das `AuthenticCommand`-Ereignis dazu verwendet werden, um jedes Mal, wenn der Benutzer auf eine Symbolleisten-Schaltfläche klickt, eine Benachrichtigung zu erhalten.

Beachten Sie bitte, dass jeder Befehl (aus der Authentic.UICommands-Sammlung) mit einem bestimmten Ereignis verknüpft ist. Um festzustellen, auf welches Symbol der Benutzer geklickt hat, muss das Skript die AuthenticEvent.srcElement-Eigenschaft überprüfen, die eine Referenz auf das entsprechende AuthenticCommand-Objekt enthält.

Beispiel

```
// event handler for OnDocEditCommand
function OnCommand()
{
 // we are interested in the k_CommandSave button
 if(objPlugIn.event.srcElement.CommandID == N)
 {
  // instead of the standard HTTP PUT we want to use
  // a HTTP POST
  objPlugIn.SavePOST();

  // no standard execution follows
  objPlugIn.event.cancelBubble = true;
 }
}
// somewhere in your script
function MyInit()
{
 objPlugIn.attachCallBack("OnDocEditCommand", OnCommand);
}
```

Referenz
Welche Befehle zur Verfügung stehen, finden Sie unter AuthenticToolbarButton.CommandID.

1.4 Ereignisse: Referenz

Liste von Connection Point Events

Name	seit TypeLib	Beschreibung
SelectionChanged	1.0	Die Auswahl im Editor hat sich geändert. Das Authentic.CurrentSelection Objekt enthält nun gültige Informationen. Die Eigenschaften des Authentic.event Objekts sind nicht definiert.
ControlInitialized	1.2	Das Control ist nun zur Gänze geladen und initialisiert. Die Eigenschaften des Authentic.event Objekts sind nicht definiert.
dragover	1.8	Eine Drag-over-Operation findet statt.
drop	1.8	Eine Drop-Operation wurde ausgeführt.
keydown	1.8	Eine Taste wurde gedrückt und noch nicht losgelassen.
keyup	1.8	Eine Taste wurde losgelassen. Dies ist normalerweise das Ereignis, das auf einen Tastendruck (keystroke) folgt.
keypressed	1.8	Wird bei jeder Eingabe über die Tastatur ausgelöst.
mousemove	1.8	Der Mauszeiger wurde bewegt.
buttonup	1.8	Eine der Maustasten wurde losgelassen.
buttondown	1.8	Eine der Maustasten wurde gedrückt.
contextmenu	1.8	Wird gesendet nach Empfang von WM_CONTEXTMENU.
editpaste	1.8	Wird aufgerufen, bevor eine Einfügeoperation durchgeführt wird.
editcut	1.8	Wird aufgerufen, bevor eine Ausschneideoperation durchgeführt wird.
editcopy	1.8	Wird aufgerufen, bevor eine Kopieroperation durchgeführt wird.
editclear	1.8	Wird aufgerufen, bevor eine Löschoperation durchgeführt wird.
doceditcommand	1.8	Wird bei Ausführung eines Authentic-Befehls und zur Implementierung eines benutzerdefinierten Befehls-Handlers ausgelöst. Siehe auch Ereignisse: Symbolleisten-Schaltflächen. Die Eigenschaften des Authentic.event Objekts sind nicht definiert.
buttondoubleclick	1.8	Doppelklick auf eine der Maustasten.

Sofern keine Ausnahme in der Beschreibung erwähnt ist, sind die Eigenschaften des Authentic.event Objekts nicht definiert.

© 2010 Altova GmbH

1.5 Aufrufen und Ändern von Dokumentinhalt

Zum Aufrufen und Ändern von Dokumentinhalt können Sie das `AuthenticRange-`, `AuthenticView-` und das `Authentic`-Objekt (und ihre Eigenschaften und Methoden) verwenden.

Wo sich die Funktionen der `AuthenticRange-` und der `AuthenticView`-Schnittstelle einerseits und der vom `Authentic`-Objekt bereitgestellten Schnittstelle auf der anderen Seite überlappen, sollte man der `AuthenticRange-` und der `AuthenticView`-Schnittstelle den Vorzug geben. Diese überlappenden Funktionalitäten des `Authentic`-Objekts werden allmählich eingestellt und in zukünftigen Versionen nicht mehr zur Verfügung stehen.

1.6 Bearbeitungsoperationen

Wenn XML-Daten in der Authentic-Ansicht angezeigt werden, können Sie einzelne Elemente mit Hilfe der Standardbearbeitungsoperationen Ausschneiden, Kopieren und Einfügen manipulieren. Es dürfen jedoch nicht alle XML-Datenelemente editiert werden. Sie müssen daher zuerst testen, ob eine Bearbeitung möglich ist. Dabei wird folgendermaßen vorgegangen: Überprüfen Sie zuerst, ob die jeweilige Bearbeitungsoperation aktiviert ist. Ist dies der Fall, rufen Sie die entsprechende Editier-Methode auf. Die einzige Methode, für die kein Test verfügbar ist, ist die `EditSelectAll`-Methode, die automatisch alle im Dokument angezeigten Elemente auswählt.

Nachstehend sehen Sie eine Liste von Eigenschaften und Methoden, die Bearbeitungsoperationen durchführen. Jede Eigenschaft gibt einen Boolschen Wert zurück. Die Methoden haben keine Parameter.

IsEditUndoEnabled	Authentic.EditUndo	Macht eine Bearbeitungsoperation rückgängig
IsEditRedoEnabled	Authentic.EditRedo	Stellt eine Bearbeitungsoperation wieder her
IsEditCopyEnabled	Authentic.EditCopy	Kopiert den markierten Text in die Zwischenablage
IsEditCutEnabled	Authentic.EditCut	Schneidet den markierten Text aus und legt ihn in die Zwischenablage
IsEditPasteEnabled	Authentic.EditPaste	Fügt den Text aus der Zwischenablage an der Cursorposition ein
IsEditClearEnabled	Authentic.EditClear	Löscht den markierten Text aus dem XML-Dokument

© 2010 Altova GmbH

1.7 Suchen und Ersetzen

Die Authentic.FindDialog-Methode öffnet ein Suchdialogfeld, in welches der Benutzer einen Suchbegriff eingeben kann. Mit Hilfe der Authentic.FindNext-Methode wird die nächste Instanz desselben Elements gesucht. Die `FindNext`-Methode kann mit Hilfe der Boolschen Eigenschaft `IsFindNextEnabled` getestet werden. Eine Variante der `FindDialog`-Methode ist die Authentic.ReplaceDialog-Methode. Mit dieser Operation wird ein bestimmtes Element gesucht und durch einen bestimmten Wert, den der Benutzer in das Ersetzungsdialogfeld eingegeben hat, ersetzt. Bei Aufruf der `FindNext`-Methode wird das nächste Element gesucht und ersetzt.

1.8 Zeilenoperationen

Sie können in der Authentic-Ansicht eine Struktur sich wiederholender Elemente als dynamische Tabelle anlegen, in der jede Zeile eine Instanz des sich wiederholenden Elements darstellt. Nachdem Sie die dynamische Tabelle erstellt haben, kann der Benutzer, der mit der Authentic-Ansicht arbeitet, die Zeilen und deren Daten manipulieren. Diese Zeilenoperationen können mit Hilfe eines Skripts ausgeführt werden.

Wenn die Zeilenoperationen von einem externen Skript ausgeführt werden sollen, sind zwei Schritte erforderlich:

- Zuerst wird überprüft, ob die Zeile, in der sich der Cursor befindet, eine Eigenschaft verwendet. Eine Eigenschaft wie z.B. `IsRowInsertEnabled` wird verwendet und gibt den Wert "True" oder "False" zurück.
- Wenn der zurückgegebene Wert "True" ist, kann die benötigte Zeilenmethode aufgerufen werden.

Im Folgenden sehen Sie eine Liste von Eigenschaften und Methoden, die Zeilenoperationen ausführen. Jede Eigenschaft gibt einen Boolschen Wert zurück und die Methoden haben keine Parameter.

IsRowInsertEnabled	Authentic.RowInsert	Zeileneinfügeoperation
IsRowAppendEnabled	Authentic.RowAppend	Operation zum Anhängen einer Zeile
IsRowDeleteEnabled	Authentic.RowDelete	Operation zum Löschen einer Zeile
IsRowMoveUpEnabled	Authentic.RowMoveUp	Verschiebt die XML-Daten um eine Zeile nach oben
IsRowMoveDownEnabled	Authentic.RowMoveDown	Verschiebt die XML-Daten um eine Zeile nach unten
IsRowDuplicateEnabled	Authentic.RowDuplicate	Dupliziert die aktuelle Zeile

© 2010 Altova GmbH

1.9 Tastaturkürzel

Falls Authentic Browser sich im Eingabemodus befindet, stehen die folgenden Tastaturkürzel zur Verfügung:

Strg + P	Dokument drucken
Strg + Z	Rückgängig
Strg + Y	Wiederherstellen
Strg + X	Ausschneiden
Strg + C	Kopieren
Strg + V	Einfügen
Strg + A	Alle auswählen
Strg + F	Suchdialogfeld öffnen
Strg + H	Ersetzungsdialogfeld öffnen

1.10 Textstatus-Schaltflächen

Die Symbolleiste der Authentic-Ansicht von Authentic Browser unterstützt Textstatussymbole. Dabei handelt es sich um Symbole, die Elemente mit vordefinierten Textformatierungseigenschaften einfügen. Um diese Funktion verwenden zu können, muss das Element, das als Textstatus-Symbol erstellt werden soll

- im Schema als globales Template deklariert sein und
- das SPS muss die erforderlichen Definitionen enthalten (siehe StyleVision -Dokumentation)

Das Plug-In benötigt den Pfad zum `.bmp`, das als Textstatus-Symbol in der Symbolleiste verwendet werden soll. Diese URL wird als der Wert des Authentic.TextStateBmpURL .-Ereignisses verwendet.

© 2010 Altova GmbH

1.11 Eingabehilfen

Authentic Browser gestattet Ihnen, Eingabehilfen wie in XMLSpy bereitzustellen. Auf diese Art hat ein Benutzer, der in der Authentic-Ansicht arbeitet, jederzeit Zugriff auf die Elemente, Attribute und Entities, die an einer bestimmten Stelle im Dokument zulässig sind. Die Eingabehilfen sind standardmäßig deaktiviert und werden nicht angezeigt, wenn sie nicht explizit aktiviert werden. Sie können die Eingabehilfen über die folgenden `PROPERTYBAG`-Parameter im `OBJECT`-Tag (Internet Explorer) oder als Attribut des `EMBED`-Elements (Firefox) aktivieren:

`EntryHelpersEnabled`	TRUE / FALSE
`EntryHelperSize`	Größe in Pixeln
`EntryHelperAlignment`	Nimmt die SPYAuthenticToolbarAllignment-Werte auf
`EntryHelperWindows`	Nimmt die SPYAuthenticEntryHelperWindows -Werte auf. Jede Kombination ist zulässig (bit-check)

Anstatt die Parameter im `OBJECT`- bzw. `EMBED`-Tag zu verwenden, können Sie auch Eigenschaften und Methoden in der API verwenden:

Eigenschaften

- Authentic.EntryHelpersEnabled
- Authentic.EntryHelperAlignment
- Authentic.EntryHelperSize
- Authentic.EntryHelperWindows

Methode

- Authentic.RedrawEntryHelpers

1.12 Pakete

Funktionsbeschreibung
Authentic Browser unterstützt Zusatzpakete, die Funktionalitäten des Programmmoduls erweitern. Derzeit bieten wir die folgenden Pakete zur Rechtschreibprüfung an. Weitere Pakete sind in Vorbereitung.

- Amerikanisches Englisch
- Englisch, komplett (einschließlich medizinisches und juristisches Englisch)
- Alle Sprachen (einschließlich medizinische und juristische Termini)

Installation der Pakete
Bei Aktualisierung eines Plug-In ist es nicht notwendig, ein bereits installiertes Paket neu zu installieren. Das bereits installierte Paket wird von der neuen Version verwendet. Das Plug-In auf dem Client-Rechner liest die XML-Einstellungsdatei (Settings File) auf dem Server und ruft die Informationen über die bereitgestellten Pakete (Versionsnummer, Beschreibung usw.) ab. Außerdem kann der Benutzer über das Paketverwaltungsdialogfeld die Verwendung der Pakete steuern.

Mechanismus
Pakete sind auf dem Server gespeichert und können vom Plug-In jederzeit geladen werden. Sobald ein Paket installiert ist, wird es bei jedem Start der Applikation aktiviert, bis es vom Benutzer entfernt wird. Der Server stellt die Pakete bereit, die bei Bedarf auf den Client heruntergeladen werden. Wenn der Benutzer das Paket nicht verwenden möchte, entfällt das weitere Download.

Aktivieren der Paketverwaltungsfunktion
Um die Paketverwaltungsfunktion zu aktivieren, verwenden Sie die `LoaderSettingsFileURL` als `PROPERTYBAG`-Parameter im `OBJECT`-Tag (Internet Explorer) oder als Attribut des `EMBED` -Elements (Firefox), die die URL der LoaderSettingsFile für die Paketverwaltung definiert.

Fügen Sie den Parameter `LoaderSettingsFileURL` bei Verwendung von Internet Explorer als Child `PARAM` Element des `OBJECT`-Elements hinzu:

```
        <PARAM NAME="LoaderSettingsFileURL" VALUE="
http://www.server.com/AuthenticFiles/XMLSpyPlugInLoaderSettings.xml "/>
```

Fügen Sie den Parameter `LoaderSettingsFileURL` bei Verwendung von Firefox als Attribut des `EMBED`-Elements hinzu:

```
        <EMBED
           ...
           LoaderSettingsFileURL="
http://www.server.com/AuthenticFiles/XMLSpyPlugInLoaderSettings.xml "/>
```

Beispiel für eine LoaderSettings XML-Datei

```
<?xml version="1.0" encoding="UTF-8"?>
<loadersettings>
   <package mode="user_demand" id="SentrySpellChecker_EAM_only"
      category="spelling" version="1">
      <packageurl>
         PlugInLoader/SentrySpellChecker_EAM_only.pck
      </packageurl>
      <description>
         This package contains the Sentry Spell Checker
         engine with an American English lexicon.
      </description>
   </package>
   <package mode="user_demand" id="SentrySpellChecker_EALL"
      category="spelling" version="1">
      <packageurl>
         PlugInLoader/SentrySpellChecker_EALL.pck
      </packageurl>
```

 © 2010 Altova GmbH

```
      <description>
         This package contains the Sentry Spell Checker engine
         with all English lexicons including Legal and Medical.
      </description>
   </package>
   <package mode="user_demand" id="SentrySpellChecker_ALL"
      category="spelling" version="1">
      <packageurl>
         PlugInLoader/SentrySpellChecker_ALL.pck
      </packageurl>
      <description>
         This package contains the Sentry Spell Checker engine
         with all lexicons including all English Legal and
         English Medical.
      </description>
   </package>
</loadersettings>
```

Beachten Sie bitte Folgendes:

- Der Pfad zum Paket muss als Inhalt des `packageurl`-Child von `package` definiert sein. Dieser Pfad kann absolut oder relativ zur HTML-Datei angegeben werden.
- Wenn Sie ein Paket aus der XML-Datei entfernen (durch Entfernen eines `package` -Elements) wird das Paket auf dem Client nicht installiert.
- Das `mode`-Attribut des `package`-Elements dient dazu, den Grad der Kontrolle festzulegen, die der Benutzer über die Entscheidung hat, ob ein Paket installiert werden soll. Bei Verwendung des Werts "user" wird der Benutzer bei jedem Start des Plug-In gefragt, ob das Paket installiert werden soll. Bei Verwendung des Werts "user_demand" wird das Paket installiert, wenn der Benutzer dies explizit verlangt. Im Fall des Rechtschreibprüfpakets wird die Installation des Pakets durch Klicken auf die Symbolleistenschaltfläche aktiviert. Bei Verwendung des Werts "force" wird das Paket installiert, ohne dass der Benutzer darüber benachrichtigt wird.
- Der Inhalt des `description` Child von `package` kann editiert werden.

1.13 Verwendung von XMLData

Mit XMLData haben Sie Zugriff auf die Elemente der aktuell angezeigten XML-Datei. Sie können damit alle erforderlichen Änderungen an den Elementen der XML-Schemastruktur vornehmen. Die Hauptfunktion von XMLData ist:

1. Zugriff auf die Namen und Werte jeder Art von Datenelementen (z.B. Elemente, Attribute)
2. Erstellung von neuen Datenelementen jeder Art
3. Einfügen und Anhängen neuer Elemente
4. Löschen vorhandener Child-Elemente

Wenn Sie mit der XMLData-Oberfläche bereits vertraut sind, da Sie sie in der XMLSpy API verwendet haben, beachten Sie bitte einige wichtige Dinge, wenn Sie neue Elemente erstellen und in die XML-Datei einfügen oder bestehende Elemente umbenennen. Lesen Sie dazu bitte die folgenden Abschnitte "Erstellen und Einfügen neuer XMLData-Objekte" und "Name und Wert von Elementen".

Struktur von XMLData
Bevor Sie die XMLData-Schnittstelle verwenden können, müssen Sie wissen, wie eine bestehende XML-Datei auf eine XMLData-Struktur gemappt wird. Beachten Sie vor allem, dass XMLData keinen eigenen Zweig von Objekten für Attribute hat.

Die Attribute eines Elements sind gleichzeitig Children des Elements. Die XMLData.Kind -Eigenschaft gibt Ihnen die Möglichkeit, zwischen den verschiedenen Child-Typen eines Elements zu unterscheiden.

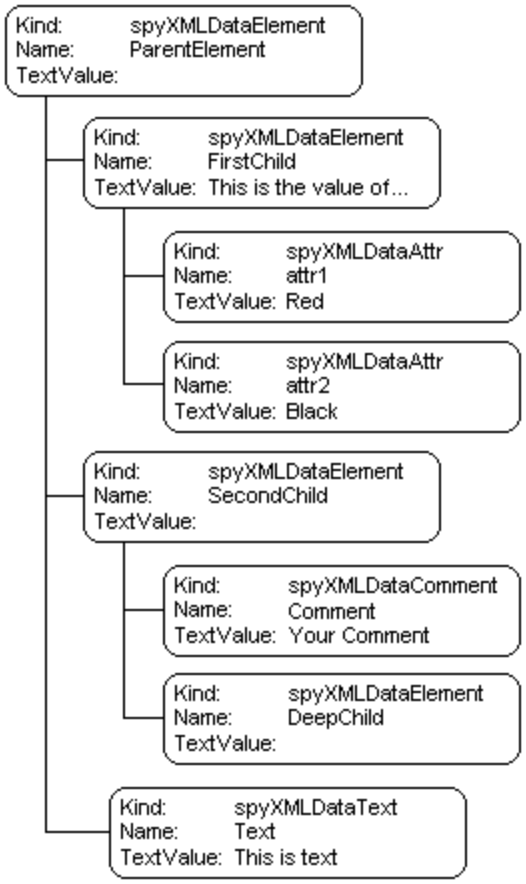
Beispiel:

Dieser XML-Code,

```
<ParentElement>
 <FirstChild attrN="Red" attr2="Black">
  This is the value of FirstChild
 </FirstChild>
 <SecondChild>
  <!--Your Comment-->
  </DeepChild>
 </SecondChild>
 This is Text
</ParentElement>
```

wird auf die folgende XMLData-Objektstruktur gemappt:

© 2010 Altova GmbH



Das Parent-Element aller XML-Elemente in einer Datei ist die Eigenschaft Authentic.XMLRoot. Über dieses XMLData-Objekt werden alle anderen XML-Elemente in der Struktur referenziert.

Name und Wert von Elementen
Mit Hilfe der Eigenschaften XMLData.Name und XMLData.TextValue können Sie den Namen und Wert aller XML-Elementtypen abrufen und ändern. Es kann vorkommen, dass mehrere Arten von XMLData-Objekten und leere Elemente keinen Textwert haben, der damit verknüpft ist.

Es ist nicht ratsam, den Namen eines bestehenden XML-Elements in der Authentic-Ansicht zu ändern. Der Name bestimmt, wie der Inhalt des Elements in der Authentic-Ansicht angezeigt wird. Nähere Informationen zu diesem Thema finden Sie in der Dokumentation zu StyleVision.

Erstellen und Einfügen neuer XMLData-Objekte
Um neue XML Language Entities einzufügen, gehen Sie folgendermaßen vor:

1. Erstellen Sie das neue XMLData-Objekt:
   Erstellen Sie mit Hilfe der Authentic.CreateChild-Methode ein neues XMLData-Objekt. Definieren Sie den Namen und Wert, bevor Sie die neue XML-Entity einfügen (siehe Punkt 3).

2. Suchen Sie den richtigen Ordner für das neue XMLData-Objekt:
   Um ein neues XMLData-Objekt einzufügen benötigen Sie zuerst eine Referenz auf das Parent-Datenelement. Wenn das neue Child-Datenelement das letzte Child-Datenelement des Parent-Datenelements werden soll, verwenden Sie zum

Einfügen des XMLData-Objekts die XMLData.AppendChild-Methode.
Falls das neue Child-Datenelement an einer anderen Stelle in der Sequenz der Child-Objekte angelegt werden soll, verwenden Sie XMLData.GetFirstChild und XMLData.GetNextChild, um den Iterator zu dem Child zu verschieben, vor dem das neue Child eingefügt werden soll.

3. Fügen Sie das neue Child mit Hilfe von XMLData.InsertChild ein. Das neue Child-Datenelement wird unmittelbar vor dem aktuellen Child-Datenelement eingefügt. Es kann in der Authentic-Ansicht vorkommen, dass zusammen mit dem erstellten Element zusätzliche Child Nodes zur XML-Datei hinzugefügt werden. Dies hängt von den Node-Einstellungen ab, die Sie über StyleVision ändern können.

Im folgenden Beispiel wird ein drittes Child-Datenelement zwischen die Elemente <FirstChild> und <SecondChild> eingefügt:

```
Dim objParent
Dim objChild
Dim objNewChild

Set objNewChild = objPlugIn.CreateChild(spyXMLDataElement)
objNewChild.Name = "OneAndAHalf"

'objParent is set to <ParentElement>
'GetFirstChild(-N) gets all children of the parent element
'and move to <SecondChild>
Set objChild = objParent.GetFirstChild(-N)
Set objChild = objParent.GetNextChild

objParent.InsertChild objNewChild
Set objNewChild = Nothing
```

Child-Elemente sollten in einer bestimmten Reihenfolge eingefügt werden. Fügen Sie nach Möglichkeit keine Attribute nach anderen Child-Elementen in die Sequenz ein. Dies bedeutet, dass Attribute keine übergeordneten Elemente eines anderen Typs haben dürfen und dass ein anderes Element kein nachfolgendes Element vom Typ Attribut haben darf.

In der Authentic-Ansicht ist eine spezielle Vorgangsweise erforderlich, um den Textwert eines XML-Elements anzuzeigen (Dies gilt nicht für Attribute). Jeder Text (oder Inhalt) muss Teil eines eigenen Child Node vom Typ Text sein. Sie können ein solches Element mit Hilfe von Authentic.CreateChild unter Verwendung des Parameterwerts 6 erstellen. Anstatt den Textwert des Elements direkt festzulegen, müssen Sie den Textwert des Child Node definieren.

Kopieren vorhandener XMLData-Objekte
Die Methoden XMLData.InsertChild und XMLData.AppendChild können nicht zum Einfügen bestehender XMLData-Objekte an einer anderen Stelle in derselben Datei verwendet werden. Diese Methoden funktionieren nur bei neuen XMLData-Objekten.

Anstatt InsertChild oder AppendChild zu verwenden, müssen Sie die Objekthierarchie manuell kopieren. Im Folgenden sehen Sie ein Beispiel für eine in JavaScript geschriebene Funktion, mit der XMLData rekursiv kopiert werden:

```
// this function returns a complete copy of the XMLData object
function GetCopy(objXMLData)
{
 var objNew;
 objNew = objPlugIn.CreateChild(objXMLData.Kind);

 objNew.Name = objXMLData.Name;
 objNew.TextValue = objXMLData.TextValue;

 if(objXMLData.HasChildren){
```


© 2010 Altova GmbH

```
  var objChild;
  objChild = objXMLData.GetFirstChild(-N);

  while(objChild) {
   try {
    objNew.AppendChild(GetCopy(objChild));
    objChild = objXMLData.GetNextChild();
   }
   catch(e) {
    objChild = null;
   }
  }
 }

 return objNew;
}
```

Löschen von XMLData-Objekten
XMLData bietet zwei Methoden zum Löschen von Child-Objekten: XMLData.EraseAllChildren und XMLData.EraseCurrentChild.

Zum Löschen von XMLData-Objekten benötigen Sie Zugriff auf das Parent-Element der zu löschenden Elemente. Verwenden Sie XMLData.GetFirstChild und XMLData.GetNextChild, um eine Referenz auf das XMLData-Parent-Objekt zu erhalten.

Beispiele zum Löschen von XML-Elementen finden Sie unter der Beschreibung der Methoden XMLData.EraseAllChildren und XMLData.EraseCurrentChild.

1.14 DOM und XMLData

Die XMLData-Schnittstelle bietet im Vergleich zu DOM weniger Methoden, ist um einiges einfacher und gibt Ihnen vollen Zugriff auf die XML-Struktur des aktuellen Dokuments. Die XMLData-Schnittstelle ist ein minimalistischer Lösungsansatz zum Lesen und Ändern bestehender oder neu erstellter XML-Daten. Sie können aber auch eine DOM-Struktur verwenden, weil Sie von einer externen Quelle darauf zugreifen können oder einfach die MSXML DOM-Implementierung bevorzugen.

Die nachfolgend beschriebenen Funktionen **ProcessDOMNode()** und **ProcessXMLDataNode()** konvertieren alle Segmente einer XML-Struktur zwischen XMLData und DOM.

So verwenden Sie die **ProcessDOMNode()**-Funktion:

- Übergeben Sie das Root-Element des DOM-Segments, das Sie in **objNode** konvertieren möchten und
- Übergeben Sie das Plug-In-Objekt mit der CreateChild()-Methode in **objCreator.**

So verwenden Sie die **ProcessXMLDataNode()**-Funktion:

- Übergeben Sie das Root-Element des XMLData-Segments in **objXMLData** und
- Übergeben Sie das mit MSXML in **xmlDoc** erzeugte DOMDocument-Objekt.

```
////////////////////////////////////////////////////////////
// DOM to XMLData conversion

function ProcessDOMNode( objNode, objCreator)
{
 var objRoot;
 objRoot = CreateXMLDataFromDOMNode( objNode, objCreator);

 if( objRoot) {
  if(( objNode. nodeValue != null) && ( objNode. nodeValue. length > 0))
   objRoot. TextValue = objNode. nodeValue;

  // add attributes
  if( objNode. attributes) {
   var Attribute;
   var oNodeList = objNode. attributes;

   for( var i = 0; i < oNodeList. length; i++) {
    Attribute = oNodeList. item( i);

    var newNode;
    newNode = ProcessDOMNode( Attribute, objCreator);

    objRoot. AppendChild( newNode);
   }
  }

  if( objNode. hasChildNodes) {
   try {
    // add children
    var Item;
    oNodeList = objNode. childNodes;

    for( var i = 0; i < oNodeList. length; i++) {
       Item = oNodeList. item( i);

     var newNode;
     newNode = ProcessDOMNode( Item, objCreator);
```

 © 2010 Altova GmbH

```
      objRoot.AppendChild(newNode);
     }
    }
    catch(err) {
    }
   }
  }

  return objRoot;
}


function CreateXMLDataFromDOMNode(objNode,objCreator)
{
 var bSetName = true;
 var bSetValue = true;

 var nKind = 4;

 switch(objNode.nodeType){
  case 2:nKind = 5;break;
  case 3:nKind = 6;bSetName = false;break;
  case 4:nKind = 7;bSetName = false;break;
  case 8:nKind = 8;bSetName = false;break;
  case 7:nKind = 9;break;
 }

 var objNew = null;
 objNew = objCreator.CreateChild(nKind);

 if(bSetName)
  objNew.Name = objNode.nodeName;

 if(bSetValue && (objNode.nodeValue != null))
  objNew.TextValue = objNode.nodeValue;

 return objNew;
}


////////////////////////////////////////////////////////////////
// XMLData to DOM conversion

function ProcessXMLDataNode(objXMLData,xmlDoc)
{
 var objRoot;
 objRoot = CreateDOMNodeFromXMLData(objXMLData,xmlDoc);

 if(objRoot) {
  if(IsTextNodeEnabled(objRoot) && (objXMLData.TextValue.length > 0))
   objRoot.appendChild(xmlDoc.createTextNode(objXMLData.TextValue));

  if(objXMLData.HasChildren){
   try {
    var objChild;
    objChild = objXMLData.GetFirstChild(-N);

    while(true) {
     if(objChild) {
      var newNode;
      newNode = ProcessXMLDataNode(objChild,xmlDoc);

      if(newNode.nodeType == 2){
       // child node is an attribute
```

```
        objRoot.attributes.setNamedItem(newNode);
       }
       else
        objRoot.appendChild(newNode);
      }

      objChild = objXMLData.GetNextChild();
     }
    }
    catch(err) {
    }
   }
  }

  return objRoot;
}

function CreateDOMNodeFromXMLData(objXMLData,xmlDoc)
{
 switch(objXMLData.Kind){
  case 4:return xmlDoc.createElement(objXMLData.Name);
  case 5:return xmlDoc.createAttribute(objXMLData.Name);
  case 6:return xmlDoc.createTextNode(objXMLData.TextValue);
  case 7:return xmlDoc.createCDATASection(objXMLData.TextValue);
  case 8:return xmlDoc.createComment(objXMLData.TextValue);
  case 9:return
xmlDoc.createProcessingInstruction(objXMLData.Name,objXMLData.TextValue);
 }

 return xmlDoc.createElement(objXMLData.Name);
}

function IsTextNodeEnabled(objNode)
{
 switch(objNode.nodeType) {
  case N:
  case 2:
  case 5:
  case 6:
  case NN:return true;
 }

 return false;
}
```


© 2010 Altova GmbH

2 Objekte

Dieser Abschnitt enthält eine Liste und Beschreibung der verschiedenen Authentic Browser Objekte.

2.1 Authentic

Siehe auch

Methoden

StartEditing
LoadXML
Reset

Save
SavePOST
SaveXML

ValidateDocument

EditClear
EditCopy
EditCut
EditPaste
EditRedo
EditSelectAll
EditUndo

RowAppend
RowDelete
RowDuplicate
RowInsert
RowMoveDown
RowMoveUp

FindDialog
FindNext
ReplaceDialog

ApplyTextState
IsTextStateApplied
IsTextStateEnabled

MarkUpView

Print
PrintPreview

CreateChild

GetAllowedElements
GetAllAttributes

© 2010 Altova GmbH

GetNextVisible
GetPreviousVisible

SelectionMoveTabOrder
SelectionSet
ClearSelection

attachCallBack

ReloadToolbars

StartSpellChecking

GetFileVersion

RedrawEntryHelpers

SetUnmodified
ClearUndoRedo

Eigenschaften
AuthenticView
IsEditClearEnabled
IsEditCopyEnabled
IsEditCutEnabled
IsEditPasteEnabled
IsEditRedoEnabled
IsEditUndoEnabled

IsFindNextEnabled

IsRowAppendEnabled
IsRowDeleteEnabled
IsRowDuplicateEnabled
IsRowInsertEnabled
IsRowMoveDownEnabled
IsRowMoveUpEnabled

SchemaLoadObject
XMLDataLoadObject
DesignDataLoadObject

XMLDataSaveUrl

XMLRoot

CurrentSelection

event

validationBadData
validationMessage

ToolbarsEnabled
ToolbarTooltipsEnabled
AutoHideUnusedCommandGroups
TextStateToolbarLine
TextStateBmpURL
ToolbarRows
UICommands
XMLTable
BaseURL

EntryHelpersEnabled
EntryHelperSize
EntryHelperAlignment
EntryHelperWindows

EnableModifications
Modified
SaveButtonAutoEnable

Ereignisse
SelectionChanged
ControlInitialized

Beschreibung
Authentic-Klasse

© 2010 Altova GmbH

2.1.1 Authentic.ApplyTextState

Siehe auch

Deklaration: ApplyTextState(*elementName* als String)

Beschreibung
Wendet den vom Parameter elementName definierten Textstatus an oder entfernt ihn. Ein typisches Beispiel für den Parameter elementName wäre "strong" oder "italic".

Ein XML-Dokument enthält Datensegmente, die Subelemente enthalten können. Nehmen wir als Beispiel den folgenden HTML-Code:

```
<b>fragment</b>
```

Aufgrund des HTML-Tags <b> wird das Wort "fragment" fett angezeigt. Dies geschieht jedoch nur deshalb, weil der HTML-Parser weiß, dass der Tag <b> "bold", also fett bedeutet. XML bietet viel mehr Flexibilität. Sie können jede beliebige Aktion für jeden beliebigen XML-Tag definieren. Wichtig ist, dass Sie mit XML einen Textstatus auf einen Text anwenden können, jedoch muss dieser Textstatus im Schema enthalten sein.

So bedeutet z.B. im Beispiel mit den Dateien OrgChart.xml OrgChart.sps, OrgChart.xsd der Tag dasselbe wie "bold". Um diesen Textstatus anzuwenden, wird die Methode **ApplyTextState()** aufgerufen. Wie auch bei den die Zeilen- und Bearbeitungsoperationen muss getestet werden, ob der Textstatus angewendet werden kann.
Siehe auch IsTextStateEnabled und IsTextStateApplied.

2.1.2 Authentic.attachCallBack

Wird nicht mehr verwendet

Verwenden Sie stattdessen die hier beschriebenen Connection Point Events.

Siehe auch

Deklaration: attachCallBack(*bstrName* als String,*varCallBack* als Variant)

Beschreibung
In der Authentic-Ansicht stehen Ereignisse zur Verfügung, die mittels benutzerdefinierter Callback-Funktionen abgearbeitet werden können. Alle Ereignishandler akzeptieren keine Parameter und alle Werte, die zurückgegeben werden, werden ignoriert. Um Informationen abzurufen, wenn ein bestimmtes Ereignis abgearbeitet wird, müssen Sie die entsprechenden Eigenschaften des event-Objekts aufrufen.

Liste der derzeit verfügbaren Ereignisse:

ondragover
ondrop
onkeydown
onkeyup
onkeypressed
onmousemove
onbuttonup
onbuttondown
oneditpaste
oneditcut
oneditcopy

Seit Version 3.0.0.0:

ondoceditcommand

Seit Version: 5.3.0.0:

onbuttondoubleclick

JavaScript-Beispiel:

```
// somwhere in your script:
objPlugIn.attachCallBack("ondragover",OnDragOver);
objPlugIn.attachCallBack("ondrop",OnDrop);

// event handlers
function OnDragOver()
{
 if(!objPlugIn.event.dataTransfer.ownDrag &&
  objPlugIn.event.dataTransfer.type == "TEXT"))
 {
  objPlugIn.event.dataTransfer.dropEffect = N;
  objPlugIn.event.cancelBubble = true;
 }
}
```


© 2010 Altova GmbH

```
// OnDrop() replaces the complete text value of the XML
// element with the selection from the drag operation
function OnDrop()
{
 var objTransfer = objPlugIn.event.dataTransfer;

 if(!objTransfer.ownDrag &&
  (objTransfer.type == "TEXT"))
  objPlugIn.event.srcElement.TextValue = objTransfer.getData();
}
```

2.1.3 AuthenticView

Siehe auch

Deklaration: AuthenticView als AuthenticView (schreibgeschützt)

Beschreibung
Gibt ein Objekt zurück, das Zugriff auf Authentic-Ansicht-spezifische Eigenschaften und Methoden verschafft.

Die AuthenticView-Funktion überlappt sich mit den bestehenden ansichtsspezifischen Funktionen. Zukünftige Versionen der Authentic-Ansicht werden alle ansichtsspezifischen Funktionen enthalten. Das AuthenticView-Objekt ist die empfohlene Schnittstelle für alle zukünftigen Implementierungen.

Beispiele
Nähere Informationen zur Verwendung des AuthenticView-Objekts finden Sie unter "Sortieren dynamischer Tabellen mit bubble-sort".

© 2010 Altova GmbH

2.1.4 Authentic.AutoHideUnusedCommandGroups

Deklaration: AutoHideUnusedCommandGroups als Boolean

Beschreibung
True, wenn nicht verwendete Symbolleisten-Befehlsgruppen automatisch ausgeblendet werden (z.B. XML-Tabellenbefehle, wenn die aktuelle SPS-Datei XML-Tabellen nicht unterstützt)
Standardwert: true

2.1.5 Authentic.BaseURL

Deklaration: BaseURL als String

Beschreibung
Diese Eigenschaft definiert eine alternative URL, die zur Auflösung von relativen Pfadangaben verwendet wird.
Wenn keine URL definiert wurde, wird der Ordner der aktuellen XML-Datei verwendet.

© 2010 Altova GmbH

2.1.6 Authentic.ClearSelection

Deklaration: ClearSelection()

Beschreibung
Mit dieser Methode wird die aktuelle Auswahl aufgehoben. Danach schlägt jeder Versuch, die Auswahl mit Hilfe der CurrentSelection-Eigenschaft zu holen, fehl, bis ein Node ausgewählt wird oder ein erfolgreicher Aufruf von SelectionSet() verarbeitet wurde.

Um einen Node löschen zu können, der die aktuelle Auswahl enthält, muss ein andere Node ausgewählt oder die Auswahl mittels ClearSelection() aufgehoben werden.

2.1.7 Authentic.ClearUndoRedo

Deklaration: ClearUndoRedo()

Beschreibung
Mit dieser Methode wird der gesamte Rückgängig/Wiederherstellen-Puffer gelöscht.

© 2010 Altova GmbH

2.1.8 Authentic.ControlInitialized

Siehe auch

Deklaration: ControlInitialized

Beschreibung
Dieses Ereignis wird bei der Erstellung und Initialisierung des Controls ausgelöst.

Siehe auch Connection Point-Ereignisse.

2.1.9 Authentic.CreateChild

Siehe auch

Deklaration: CreateChild(*nKind* als SPYXMLDataKind) als XMLData

Rückgabewert
Neuer XML Node

Beschreibung

Die CreateChild-Methode dient zur Erstellung neuer Nodes, die Sie über die XMLData -Schnittstelle in die XML-Struktur des aktuellen Dokuments einfügen können.

Siehe auch XMLData.AppendChild und XMLData.InsertChild

© 2010 Altova GmbH

2.1.10 Authentic.CurrentSelection

Siehe auch

Deklaration: CurrentSelection als AuthenticSelection

Beschreibung
Die Eigenschaft verschafft Zugriff auf die aktuelle Auswahl in der Authentic-Ansicht.

Mit dem unten gezeigten Beispielcode wird der komplette Text der aktuellen Auswahl abgerufen:

JavaScript:

```
// somewhere in your script:
GetSelection(objPlugIn.CurrentSelection);

// GetSelection() collects complete text selection
function GetSelection(objSel)
{
 var strText = "";

 var objCurrent = objSel.Start;

 while(!objSel.End.IsSameNode(objCurrent))
 {
  objCurrent = objPlugIn.GetNextVisible(objCurrent);
  strText += objCurrent.TextValue;
 }

 strText += objSel.End.TextValue.substring(0,objSel.EndTextPosition);
 return objSel.Start.TextValue.substr(objSel.StartTextPosition) + strText;
}
```

2.1.11 Authentic.DesignDataLoadObject

Siehe auch

Deklaration: DesignDataLoadObject als AuthenticLoadObject

Beschreibung
Das DesignDataLoadObject enthält eine Referenz auf das SPS-Dokument. Mit Hilfe des SPS-Dokuments, das normalerweise in StyleVision erstellt wird, wird die WYSIWYG-Editierumgebung erstellt.

Ein Beispiel dazu finden Sie unter SchemaLoadObject.

© 2010 Altova GmbH

2.1.12 Authentic.EditClear

Siehe auch

Deklaration: EditClear

Beschreibung
Hebt die aktuelle Auswahl auf.

2.1.13 Authentic.EditCopy

Siehe auch

Deklaration: EditCopy

Beschreibung
Kopiert die aktuelle Auswahl in die Zwischenablage.

© 2010 Altova GmbH

2.1.14 Authentic.EditCut

Siehe auch

Deklaration: EditCut

Beschreibung
Schneidet die aktuelle Auswahl aus dem Dokument aus und kopiert sie in die Zwischenablage.

2.1.15 Authentic.EditPaste

Siehe auch

Deklaration: EditPaste

Beschreibung
Fügt den Inhalt aus der Zwischenablage in das Dokument ein.

© 2010 Altova GmbH

2.1.16 Authentic.EditRedo

Siehe auch

Deklaration: EditRedo

Beschreibung
Stellt die zuletzt rückgängig gemachte Aktion wieder her.

2.1.17 Authentic.EditSelectAll

Siehe auch

Deklaration: EditSelectAll

Beschreibung
Mit dieser Methode wird das komplette Dokument ausgewählt.

© 2010 Altova GmbH

2.1.18 Authentic.EditUndo

Siehe auch

Deklaration: EditUndo

Beschreibung
Die letzte Aktion wird rückgängig gemacht.

2.1.19 Authentic.EnableModifications

Deklaration: EnableModifications als Boolean

Beschreibung
"True", wenn die durch Authentic Browser am XML-Inhalt vorgenommenen Änderungen aktiviert sind. Siehe auch Modified-Eigenschaft.

Standardwert: TRUE

© 2010 Altova GmbH

2.1.20 Authentic.EntryHelperAlignment

Deklaration: EntryHelperAlignment als SPYAuthenticToolbarAllignment

Beschreibung
Diese Eigenschaft dient zum Festlegen der Position der Eingabehilfen. Der Standardwert ist 3. Damit werden die Eingabehilfen an der rechten Seite platziert.

2.1.21 Authentic.EntryHelpersEnabled

Deklaration: EntryHelpersEnabled als Boolean

Beschreibung
"True", wenn die Eingabehilfen von Authentic Browser aktiviert sind.
Diese Eigenschaft kann verwendet werden, um die Eingabehilfen zu aktivieren bzw. zu deaktivieren.

Standardwert: FALSE

© 2010 Altova GmbH

2.1.22 Authentic.EntryHelperSize

Deklaration: EntryHelperSize als Integer

Beschreibung
Mit Hilfe dieser Eigenschaft können Sie die anfangs verwendete Größe der Eingabehilfen in Pixeln angeben. Setzen Sie die Eigenschaft auf -1, wenn dieser Wert ignoriert werden soll.

Standardwert: -1

2.1.23 Authentic.EntryHelperWindows

Deklaration: EntryHelperWindows als SPYAuthenticEntryHelperWindows

Beschreibung
Mit Hilfe dieser Eigenschaft können Sie definieren, welche der Eingabehilfen angezeigt werden sollen. Die Werte können kombiniert werden, um mehrere Eingabehilfenfenster anzuzeigen. Der Standardwert ist 7. Bei Auswahl dieses Werts werden alle drei Eingabehilfen eingeblendet.

© 2010 Altova GmbH

2.1.24 Authentic.event

Siehe auch

Deklaration: event als AuthenticEvent

Beschreibung
Die Ereigniseigenschaft enthält ein Ereignis-Datenobjekt mit Informationen über das aktuelle Ereignis.

2.1.25 Authentic.FindDialog

Siehe auch

Deklaration: FindDialog

Beschreibung
Zeigt den FindDialog an.

Siehe auch Suchen und Ersetzen.

© 2010 Altova GmbH

2.1.26 Authentic.FindNext

Siehe auch

Deklaration: FindNext

Beschreibung
Diese Methode führt eine "Weitersuchen"-Operation aus.

Siehe auch Suchen und Ersetzen.

2.1.27 Authentic.GetAllAttributes

Siehe auch

Deklaration: GetAllAttributes(*pForElement* als XMLData, *pElements* als Variant)

Beschreibung
GetAllAttributes() gibt die für das angegebene Element zulässigen Attribute als String-Array zurück.

JavaScript-Beispiel:

```
function GetAttributes()
{
 var arrElements = new Array(N);

 var objStart = objPlugIn.CurrentSelection.Start;

 var strText;
 strText = "Valid attributes at current selection:\n\n";

 for(var i = N;i <= 4;i++)
 {
  objPlugIn.GetAllAttributes(objStart, arrElements);
  strText = strText + ListArray(arrElements) + "--------------------\n";
 }

 return strText;
}

function ListArray(arrIn)
{
 var strText = "";

 if(typeof(arrIn) == "object")
 {
  for(var i = 0;i <= (arrIn.length - N);i++)
   strText = strText + arrIn[i] + "\n";
 }

 return strText;
}
```

VBScript-Beispiel:

```
Sub DisplayAllowedAttributes
 dim arrElements()

 dim objStart
 dim objEnd
 set objStart = objPlugIn.CurrentSelection.Start
 set objEnd = objPlugIn.CurrentSelection.End

 dim strText
 strText = "Valid attributes at current selection:" & chr(N3) & chr(N3)

 dim i

 For i = N To 4
  objView.GetAllAttributes objStart, arrElements
  strText = strText & ListArray(arrElements) & "----------------" & chr(N3)
 Next
```


© 2010 Altova GmbH

```
 msgbox strText
End Sub

Function ListArray( arrIn)
 dim strText

 If IsArray( arrIn) Then
  dim i

  For i = 0 To UBound( arrIn)
   strText = strText & arrIn( i) & chr( N3)
  Next
 End If

 ListArray = strText
End Function
```

2.1.28 Authentic.GetAllowedElements

Siehe auch

Deklaration: GetAllowedElements(*nAction* als SPYAuthenticElementActions, *pStartElement* als XMLData,*pEndElement* als XMLData,*pElements* als Variant)

Beschreibung
GetAllowedElements() gibt die für die verschiedenen von nAction definierten Aktionen zulässigen Elemente zurück.

JavaScript-Beispiel:

```
function GetAllowed()
{
 var arrElements = new Array(N);

 var objStart = objPlugIn.CurrentSelection.Start;
 var objEnd = objPlugIn.CurrentSelection.End;

 var strText;
 strText = "valid elements at current selection:\n\n";

 for(var i = 0;i <= 4;i++){
  objPlugIn.GetAllowedElements(i,objStart,objEnd,arrElements);
  strText = strText + ListArray(arrElements) + "------------------\n";
 }

 return strText;
}

function ListArray(arrIn)
{
 var strText = "";

 if(typeof(arrIn) == "object"){
  for(var i = 0;i <= (arrIn.length - N);i++)
   strText = strText + arrIn[i] + "\n";
 }

 return strText;
}
```

VBScript-Beispiel:

```
Sub DisplayAllowed
 dim arrElements()

 dim objStart
 dim objEnd
 set objStart = objPlugIn.CurrentSelection.Start
 set objEnd = objPlugIn.CurrentSelection.End

 dim strText
 strText = "Valid elements at current selection:" & chr(N3) & chr(N3)

 dim i

 For i = N To 4
  objView.GetAllowedElements i,objStart,objEnd,arrElements
  strText = strText & ListArray(arrElements) & "----------------" & chr(N3)
 Next
```


© 2010 Altova GmbH

```
 msgbox strText
End Sub

Function ListArray( arrIn)
 dim strText

 If IsArray( arrIn) Then
  dim i

  For i = 0 To UBound( arrIn)
   strText = strText & arrIn( i) & chr( N3)
  Next
 End If

 ListArray = strText
End Function
```

2.1.29 Authentic.GetFileVersion

Siehe auch

Deklaration: GetFileVersion(*strVersion* als String)

Beschreibung
Die Methode gibt die Version der Komponente als String im Format 5.0.0.0 zurück.

© 2010 Altova GmbH

2.1.30 Authentic.GetNextVisible

Siehe auch

Deklaration: GetNextVisible(*pElement* als XMLData) als XMLData

Beschreibung
Die Methode liefert das nächste sichtbare XML-Element im Dokument.

2.1.31 Authentic.GetPreviousVisible

Siehe auch

Deklaration: GetPreviousVisible(*pElement* als XMLData) als XMLData

Beschreibung
Die Methode liefert das vorhergehende XML-Element im Dokument.

© 2010 Altova GmbH

2.1.32 Authentic.IsEditClearEnabled

Siehe auch

Deklaration: IsEditClearEnabled als Boolean

Beschreibung
True, wenn EditClear möglich ist.

Siehe auch Bearbeitungsoperationen.

2.1.33 Authentic.IsEditCopyEnabled

Siehe auch

Deklaration: IsEditCopyEnabled als Boolean

Beschreibung
True, wenn Kopieren in die Zwischenablage möglich ist.

Siehe auch EditCopy und Bearbeitungsoperationen.

© 2010 Altova GmbH

2.1.34 Authentic.IsEditCutEnabled

Siehe auch

Deklaration: IsEditCutEnabled als Boolean

Beschreibung
True, wenn EditCut derzeit möglich ist.

Siehe auch Bearbeitungsoperationen.

2.1.35 Authentic.IsEditPasteEnabled

Siehe auch

Deklaration: IsEditPasteEnabled als Boolean

Beschreibung
True, wenn EditPaste möglich ist.

Siehe auch Bearbeitungsoperationen.

© 2010 Altova GmbH

2.1.36 Authentic.IsEditRedoEnabled

Siehe auch

Deklaration: IsEditRedoEnabled als Boolean

Beschreibung
True, wenn EditRedo derzeit möglich ist.

Siehe auch Bearbeitungsoperationen.

2.1.37 Authentic.IsEditUndoEnabled

Siehe auch

Deklaration: IsEditUndoEnabled als Boolean

Beschreibung
True, wenn EditUndo möglich ist.

Siehe auch Bearbeitungsoperationen.

© 2010 Altova GmbH

2.1.38 Authentic.IsFindNextEnabled

Siehe auch

Deklaration: IsFindNextEnabled als Boolean

Beschreibung
True, wenn FindNext derzeit möglich ist. False, wenn der Suchbegriff nicht mehr vorkommt.

Siehe auch Suchen und Ersetzen und FindDialog.

2.1.39 Authentic.IsRowAppendEnabled

Siehe auch

Deklaration: IsRowAppendEnabled als Boolean

Beschreibung
True, wenn RowAppend möglich ist.

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.40 Authentic.IsRowDeleteEnabled

Siehe auch

Deklaration: IsRowDeleteEnabled als Boolean

Beschreibung
True, wenn RowDelete möglich ist.

Siehe auch Zeilenoperationen.

2.1.41 Authentic.IsRowDuplicateEnabled

Siehe auch

Deklaration: IsRowDuplicateEnabled als Boolean

Beschreibung
True, wenn RowDuplicate möglich ist.

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.42 Authentic.IsRowInsertEnabled

Siehe auch

Deklaration: IsRowInsertEnabled als Boolean

Beschreibung
True, wenn RowInsert möglich ist.

Siehe auch Zeilenoperationen.

2.1.43 Authentic.IsRowMoveDownEnabled

Siehe auch

Deklaration: IsRowMoveDownEnabled als Boolean

Beschreibung
True, wenn RowMoveDown möglich ist.

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.44 Authentic.IsRowMoveUpEnabled

Siehe auch

Deklaration: IsRowMoveUpEnabled als Boolean

Beschreibung
True, wenn RowMoveUp möglich ist.

Siehe auch Zeilenoperationen.

2.1.45 Authentic.IsTextStateApplied

Siehe auch

Deklaration: IsTextStateApplied(*elementName* als String) als Boolean

Beschreibung
Überprüft, ob der Textstatus bei der aktuellen Selektion/Position bereits angewendet wurde. Typische Beispiele für den Parameter elementName sind "strong" und "italic".

© 2010 Altova GmbH

2.1.46 Authentic.IsTextStateEnabled

Siehe auch

Deklaration: IsTextStateEnabled(*elementName* als String) als Boolean

Beschreibung
Überprüft, ob ein Textstatus für die aktuelle Selektion/Position angewendet werden kann. Typische Beispiele für den Parameter elementName sind "strong" und "italic".

2.1.47 Authentic.LoadXML

Siehe auch

Deklaration: LoadXML(*xmlString* als String)

Beschreibung
Lädt den angegebenen String als das aktuelle XML-Dokument in die Authentic-Ansicht. Der neue Inhalt wird sofort angezeigt.

© 2010 Altova GmbH

2.1.48 Authentic.MarkUpView

Siehe auch

Deklaration: MarkUpView(*kind* als SPYAuthenticMarkupVisibility)

Beschreibung
Standardmäßig wird das Dokument wie ein HTML-Dokument angezeigt. Manchmal ist es jedoch hilfreich, wenn die Editier-Tags angezeigt werden. Mit Hilfe dieser Methode können Sie verschiedene Arten von Markup-Tags anzeigen.

2.1.49 Authentic.Modified

Deklaration: Modified als Boolean

Beschreibung
True, wenn XML-Inhalt geändert wurde.

Diese Eigenschaft ist schreibgeschützt.

© 2010 Altova GmbH

2.1.50 Authentic.Print

Siehe auch

Deklaration: Print

Beschreibung
Druckt das aktuell editierte Dokument.

2.1.51 Authentic.PrintPreview

Siehe auch

Deklaration: PrintPreview

Beschreibung
Zeigt eine Druckvorschau des gerade editierten Dokuments an.

© 2010 Altova GmbH

2.1.52 Authentic.RedrawEntryHelpers

Deklaration: RedrawEntryHelpers()

Beschreibung
RedrawEntryHelpers holt die Werte aus den Eigenschaften EntryHelpersEnabled, EntryHelperAlignment, EntryHelperSize und EntryHelperWindows und zeichnet die Eingabehilfenfenster neu.

2.1.53 Authentic.ReloadToolbars

Deklaration: ReloadToolbars()

Beschreibung
ReloadToolbars liest die ToolbarRows Sammlung und zeichnet die Symbolleisten und Ansichten neu.

© 2010 Altova GmbH

2.1.54 Authentic.ReplaceDialog

Siehe auch

Deklaration: ReplaceDialog

Beschreibung
Zeigt den ReplaceDialog an.

Siehe auch Suchen und Ersetzen.

2.1.55 Authentic.Reset

Veraltet

Verwenden Sie statt dessen Authentic.StartEditing.

Siehe auch

Deklaration: Reset

Beschreibung
Wurde durch andere Funktionen ersetzt und es wird angeraten diese nicht mehr zu verwenden.

© 2010 Altova GmbH

2.1.56 Authentic.RowAppend

Siehe auch

Deklaration: RowAppend

Beschreibung
Hängt eine Zeile an der aktuellen Position an.

Siehe auch Zeilenoperationen.

2.1.57 Authentic.RowDelete

Siehe auch

Deklaration: RowDelete

Beschreibung
Löscht die aktuell ausgewählte(n) Zeile(n).

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.58 Authentic.RowDuplicate

Siehe auch

Deklaration: RowDuplicate

Beschreibung
Diese Methode dupliziert die aktuell ausgewählten Zeilen.

Siehe auch Zeilenoperationen.

2.1.59 Authentic.RowInsert

Siehe auch

Deklaration: RowInsert

Beschreibung
Fügt eine neue Zeile unmittelbar oberhalb der aktuellen Auswahl ein.

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.60 Authentic.RowMoveDown

Siehe auch

Deklaration: RowMoveDown

Beschreibung
Verschiebt die aktuelle Zeile um eine Position nach unten.

Siehe auch Zeilenoperationen.

2.1.61 Authentic.RowMoveUp

Siehe auch

Deklaration: RowMoveUp

Beschreibung
Verschiebt die aktuelle Zeile um eine Position nach oben.

Siehe auch Zeilenoperationen.

© 2010 Altova GmbH

2.1.62 Authentic.Save

Siehe auch

Deklaration: Save

Beschreibung
Speichert das Dokument unter der in der Eigenschaft XMLDataSaveUrl definierten URL. Bei Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad verwenden.

Das Plug-In sendet einen HTTP PUT Request an den Server, um die aktuell angezeigte XML-Datei zu speichern.

2.1.63 Authentic.SaveButtonAutoEnable

Deklaration: SaveButtonAutoEnable als Boolean

Beschreibung
Ist diese Eigenschaft auf TRUE gesetzt, wird der Ein/Aus-Status der Schaltfläche "Speichern" in der Symbolleiste des Control entsprechend dem Modified Flag des Dokuments gesetzt.

Der Standardwert ist FALSE.

© 2010 Altova GmbH

2.1.64 Authentic.SavePOST

Siehe auch

Deklaration: SavePOST

Beschreibung
Speichert das Dokument unter der von der Eigenschaft XMLDataSaveUrl definierten URL. Bei Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad verwenden. Das Plug-In sendet einen HTTP POST Request an den Server, um die aktuell angezeigte XML-Datei zu speichern.

Überprüfen, ob eine Datei gespeichert wurde oder nicht
Wenn das Authentic Plug-In eine HTTP-Antwort `>= 300` erhält, geht es davon aus, dass die Datei **nicht** gespeichert wurde und zeigt daraufhin (standardmäßig) eine Meldung an, die den HTTP-Fehler enthält. Anschließend wird eine zweite (deaktivierbare) Meldung angezeigt, in der der Benutzer darauf hingewiesen wird, dass die Datei nicht gespeichert wurde. Es obliegt dem Applikationsentwickler, sicherzustellen, dass die korrekte HTTP-Antwort zurückgegeben wird, je nachdem ob das Dokument gespeichert werden konnte oder nicht. Der entsprechende PHP-Code könnte etwa folgendermaßen aussehen:

```
<?php
// suppress error messages to prevent any output
// being generated before headers can be sent
error_reporting (0);
$error = false;
$handle = fopen ( "result.xml", "w+" );
if (! $handle)
   $error = true;
else
{
   if (! fwrite($handle, $HTTP_RAW_POST_DATA))
          $error = true;
   else
          fclose($handle);
}
if ($error)
   header( "HTTP/N.N 500 Server Error" );
?>
```

2.1.65 Authentic.SaveXML

Siehe auch

Deklaration: SaveXML als String

Rückgabewert
XML-Struktur als String

Beschreibung
Speichert die aktuellen XML-Daten in einem String, der an den Caller zurückgegeben wird.

© 2010 Altova GmbH

2.1.66 Authentic.SchemaLoadObject

Siehe auch

Deklaration: SchemaLoadObject als AuthenticLoadObject

Beschreibung
Das SchemaLoadObject enthält eine Referenz zum XML-Schema-Dokument für die aktuelle XML-Datei. Das Schema-Dokument wird normalerweise mit XMLSpy generiert.

Beispiel

```
objPlugIn.SchemaLoadObject.URL = "http://www.YOURSERVER.com/OrgChart.xsd"
objPlugIn.XMLDataLoadObject.URL = "http://www.YOURSERVER.com/OrgChart.xml"
objPlugIn.DesignDataLoadObject.URL = "http://www.YOURSERVER.com/OrgChart.sps"
objPlugIn.StartEditing
```

Der obige Code definiert alle URL-Eigenschaften der zu ladenden Objekte und ruft StartEditing zum Laden und Anzeigen der Dateien auf. Bei Untrusted Versionen können Sie auch einen vollständigen lokalen Pfad verwenden. Der aktuelle Inhalt und Status des Plug-In werden zurückgesetzt.

2.1.67 Authentic.SelectionChanged

Siehe auch

Deklaration: SelectionChanged als VT_0019

Beschreibung
Dieses Ereignis wird ausgelöst, wenn der Benutzer die aktuelle Auswahl ändert.

Siehe auch Connection Point-Ereignisse.

© 2010 Altova GmbH

2.1.68 Authentic.SelectionMoveTabOrder

Siehe auch

Deklaration: SelectionMoveTabOrder(*bForward* als Boolean,*bTag* als Boolean)

Beschreibung
SelectionMoveTabOrder() verschiebt die aktuelle Auswahl nach vorne oder nach hinten.

Wenn bTag "false" ist und die letzte Zelle einer Tabelle ausgewählt ist, wird eine neue Zeile hinzugefügt.

2.1.69 Authentic.SelectionSet

Siehe auch

Deklaration: SelectionSet(*pStartElement* als XMLData,*nStartPos* als long, *pEndElement* als XMLData,*nEndPos* als long) als Boolean

Beschreibung
Verwenden Sie SelectionSet(), um eine neue Auswahl in der Authentic-Ansicht zu definieren. Sie können pEndElement auf Null (nichts) setzen, wenn die Auswahl nicht mehr als ein (pStartElement) XML-Element umfassen soll.

© 2010 Altova GmbH

2.1.70 Authentic.SetUnmodified

Deklaration: SetUnmodified()

Beschreibung
Nach Aufruf dieser Methode wird der aktuelle Zustand der Rückgängig/Wiederholen-Puffers als ungeänderter Zustand des zugrunde liegenden XML-Dokuments verwendet und der Modified Flag wird auf FALSE gesetzt.

2.1.71 Authentic.StartEditing

Siehe auch

Deklaration: StartEditing als Boolean

Rückgabewert
True, wenn alle Dateien erfolgreich geladen wurden und angezeigt werden.

Beschreibung
Startet die Editierung des aktuellen Dokuments. Es ist wichtig, dass zuerst die Eigenschaften der Lade-Objekte SchemaLoadObject, DesignDataLoadObject und XMLDataLoadObject festgelegt werden.

© 2010 Altova GmbH

2.1.72 Authentic.StartSpellChecking

Deklaration: StartSpellChecking()

Beschreibung
Dieser Befehl öffnet das Dialogfeld für die Rechtschreibprüfung, wenn ein Paket mit dem Rechtschreibprüf-Prozessor und den erforderlichen Wörterbüchern verfügbar und aktiviert ist.

2.1.73 Authentic.TextStateBmpURL

Deklaration: TextStateBmpURL als String

Beschreibung
Die URL, von der die Bitmaps für die Textstatus-Symbole abgerufen werden sollen. Wenn keine URL definiert ist, werden keine Textstatus-Symbole angezeigt.

Beispiele
objPlugIn.TextStateBmpURL = "<http://plugin.xmlspy.com/textstates/>"

```
<PARAM NAME="TextStateBmpURL" VALUE="http://plugin.xmlspy.com/textstates/">
```


© 2010 Altova GmbH

2.1.74 Authentic.TextStateToolbarLine

Deklaration: TextStateToolbarLine als long

Beschreibung
Die Symbolleiste (Zeilennummer) in der die Textstatus-Symbole angezeigt werden sollen. Textstatus-Symbole können an bestehende Symbolleisten angehängt oder in neue Symbolleisten platziert werden.
Standardwert: 1

2.1.75 Authentic.ToolbarRows

Deklaration: ToolbarRows als AuthenticToolbarRows

Beschreibung
Enthält eine Sammlung aller Symbolleisten, die angezeigt werden sollen. Informationen zum Löschen, Hinzufügen oder Ändern von Symbolleisten finden Sie unter AuthenticToolbarRows.

© 2010 Altova GmbH

2.1.76 Authentic.ToolbarsEnabled

Deklaration: ToolbarsEnabled als Boolean

Beschreibung
True, wenn Authentic Browser-Symbolleisten aktiviert sind.
Diese Eigenschaft kann zum Aktivieren oder Deaktivieren aller Symbolleisten verwendet werden.
Standardwert: true

2.1.77 Authentic.ToolbarTooltipsEnabled

Deklaration: ToolbarTooltipsEnabled als Boolean

Beschreibung
True, wenn die Tooltipps für die Authentic Browser-Symbolleisten aktiviert sind.
Standardwert: true

© 2010 Altova GmbH

2.1.78 Authentic.UICommands

Deklaration: UICommands als AuthenticCommands

Beschreibung
Enthält eine Sammlung aller verfügbaren Symbolleistenbefehle (mit Beschreibung).
Für diese Befehle können in den verschiedenen Symbolleisten Schaltflächen platziert werden.
Schreibgeschützt.

Beispiel
Holt alle verfügbaren Befehle, Befehlsgruppen, Beschreibungen und zeigt Sie in einem Nachrichtenfeld an.

```
dim str
for each UICommand in objPlugin.UICommands
str = str & UICommand.CommandID & " | " & UICommand.Group & " | " &
UICommand.ShortDescription & chr(N3)
next
msgbox str
```

2.1.79 Authentic.ValidateDocument

Siehe auch

Deklaration: ValidateDocument(*showResults* als Boolean) als Boolean

Rückgabewert
Ergebnis der Validierung

Beschreibung
Überprüft die aktuellen XML-Daten auf Ihre Gültigkeit hinsichtlich der XML-Schemadaten. Wenn der Parameter showResults FALSE ist, werden die Validierungsfehler nicht angezeigt, ansonsten werden sie angezeigt.

© 2010 Altova GmbH

2.1.80 Authentic.validationBadData

Siehe auch

Deklaration: validationBadData als XMLData

Beschreibung
Diese Eigenschaft kann zusätzliche Informationen über den letzten Validierungsfehler liefern. Sie wird nach Aufrufen von `ValidateDocument()` gesetzt und ist entweder Null oder sie enthält eine Referenz auf das fehlerverursachende XML-Element.

2.1.81 Authentic.validationMessage

Siehe auch

Deklaration: validationMessage als String

Beschreibung
Wenn die Validierung fehlgeschlagen ist (nach Aufruf von ValidateDocument), speichert diese Eigenschaft einen String mit der Fehlermeldung.

© 2010 Altova GmbH

2.1.82 Authentic.XMLDataLoadObject

Siehe auch

Deklaration: XMLDataLoadObject als AuthenticLoadObject

Beschreibung
Das XMLDataLoadObject enthält eine Referenz auf das XML-Dokument, das gerade bearbeitet wird. Das XML-Dokument wird normalerweise mit Hilfe von XMLSpy definiert und anhand einer Datenbank oder eines anderes Geschäftsprozesses generiert.

Beispiel dazu finden Sie unter SchemaLoadObject.

2.1.83 Authentic.XMLDataSaveUrl

Siehe auch

Deklaration: XMLDataSaveUrl als String

Beschreibung
Wenn die XML-Daten geändert wurden, können diese über eine URL wieder auf einem Server gespeichert werden. Bei Speichern der XML-Daten mittels eines `HTTP PUT` mit der Methode `Authentic.Save` wird mit dieser Eigenschaft der Pfad definiert, unter dem die XML-Daten gespeichert werden. Wenn Sie die Daten mit der `Authentic.SavePOST` Methode über ein `HTTP POST`absenden, wird mit dieser Eigenschaft der Pfad des serverseitigen Scripts bzw. der serverseitigen Applikation definiert, das/die die `POST`-Daten verarbeiten soll. Bei Untrusted Versionen können Sie auch den vollständigen lokalen Pfad verwenden.

Siehe auch die Methoden Authentic.Save und Authentic.SavePOST.

© 2010 Altova GmbH

2.1.84 Authentic.XMLRoot

Siehe auch

Deklaration: XMLRoot als XMLData

Beschreibung
XMLRoot ist das Parent-Element der aktuell angezeigten XML-Struktur. Über die XMLData Schnittstelle haben Sie vollen Zugriff auf den kompletten Dateiinhalt.

Weitere Informationen dazu siehe auch Using XMLData.

2.1.85 Authentic.XMLTable

Deklaration: XMLTable als AuthenticXMLTableCommands

Beschreibung
Holt einen Satz aller XML-Tabellenbefehle.
Schreibgeschützt.

© 2010 Altova GmbH

2.2 AuthenticCommand

Methoden

Eigenschaften

CommandID
Group
ShortDescription
Name

2.2.1 AuthenticCommand.CommandID

Deklaration: CommandID als SPYAuthenticCommand

Beschreibung
Enthält die CommandId des Befehls
Mögliche Werte: siehe AuthenticToolbarButton
Schreibgeschützt

Beispiel
Siehe Beispiel unter Authentic.UICommands

© 2010 Altova GmbH

2.2.2 AuthenticCommand.Group

Deklaration: Group als SPYAuthenticCommandGroup

Beschreibung
Die CommandGroup, zu der der Befehl gehört.
Schreibgeschützt

Beispiel
Siehe Beispiel unter Authentic.UICommands

2.2.3 AuthenticCommand.ShortDescription

Deklaration: ShortDescription als String

Beschreibung
Kurzbeschreibung des Befehls (z.B. der Tooltipp-Text)
Schreibgeschützt

Beispiel
Siehe Beispiel unter Authentic.UICommands

© 2010 Altova GmbH

2.2.4 AuthenticCommand.Name

Deklaration: Name als String

Beschreibung
Für zukünftige Verwendungszwecke

2.3 AuthenticCommands

Methoden

Item

Eigenschaften

Count

© 2010 Altova GmbH

2.3.1 AuthenticCommands.Count

Deklaration: Count als long

Beschreibung
Anzahl der verfügbaren Benutzerschnittstellen-Befehle.
Schreibgeschützt

2.3.2 AuthenticCommands.Item

Deklaration: Item (nPosition als long) als AuthenticCommand

Beschreibung
Liefert den Befehl in Position nPosition. nPosition beginnt mit 1.

© 2010 Altova GmbH

2.4 AuthenticDataTransfer

Siehe auch

Methoden
getData

Eigenschaften
dropEffect
ownDrag
type

Beschreibung
AuthenticDataTransfer-Schnittstelle.

2.4.1 AuthenticDataTransfer.dropEffect

Siehe auch

Deklaration: dropEffect als long

Beschreibung
Die Eigenschaft speichert das Drop-Verhalten aus dem Standard-Ereignishandler. Sie können das Drop-Verhalten definieren, wenn Sie diesen Wert ändern und AuthenticEvent.cancelBubble auf TRUE setzen.

© 2010 Altova GmbH

2.4.2 AuthenticDataTransfer.getData

Siehe auch

Deklaration: getData als Variant

Beschreibung
getData holt die aktuellen Daten, die mit diesem dataTransfer-Objekt verknüpft sind. Siehe auch AuthenticDataTransfer.type.

2.4.3 AuthenticDataTransfer.ownDrag

Siehe auch

Deklaration: ownDrag als Boolean

Beschreibung
Die Eigenschaft ist TRUE, wenn die Quelle des Drag-Vorgangs von innerhalb der Authentic -Ansicht kommt.

© 2010 Altova GmbH

2.4.4 AuthenticDataTransfer.type

Siehe auch

Deklaration: type als String

Beschreibung
Enthält den Datentyp, den Sie mit der AuthenticDataTransfer.getData Methode holen.

Derzeit werden die folgenden Datentypen unterstützt:

OWN	Daten aus dem Plug-In selbst
TEXT	Nur Text
UNICODETEXT	Nur Text als UNICODE
IUNKNOWN	Unbekannte Referenz auf IDataObject-Instanz

2.5 AuthenticEvent

Siehe auch

Eigenschaften
altKey
altLeft
ctrlKey
ctrlLeft
shiftKey
shiftLeft

keyCode
repeat

button

clientX
clientY

dataTransfer

srcElement
fromElement

propertyName

cancelBubble
returnValue

type

Beschreibung
AuthenticEvent-Schnittstelle.

© 2010 Altova GmbH

2.5.1 AuthenticEvent.altKey

Siehe auch

Deklaration: altKey als Boolean

Beschreibung
True, wenn Sie die rechte ALT-Taste drücken.

2.5.2 AuthenticEvent.altLeft

Siehe auch

Deklaration: altLeft als Boolean

Beschreibung
True, wenn die linke ALT-Taste gedrückt wird.

© 2010 Altova GmbH

2.5.3 AuthenticEvent.button

Siehe auch

Deklaration: button als long

Beschreibung
Gibt an, welche Maustaste gedrückt wird:

0	Keine Taste
1	Linke Taste
2	Rechte Taste
3	Sowohl die linke als auch die rechte Taste
4	Mittlere Taste
5	Linke und mittlere Taste
6	Rechte und mittlere Taste
7	Alle drei Tasten

Die Ereignisse onbuttondown und onbuttonup setzen den Tastenwert auf verschiedene Arten. D as onbuttonup-Ereignis setzt immer nur den Wert für die Taste, die gerade losgelassen wurde und das Ereignis ausgelöst hat, unabhängig davon, welche anderen Tasten in diesem Moment ebenfalls gedrückt sind.

2.5.4 AuthenticEvent.cancelBubble

Siehe auch

Deklaration: cancelBubble als Boolean

Beschreibung
Setzt cancelBubble auf TRUE, wenn der Standard-Ereignishandler nicht aufgerufen werden soll.

© 2010 Altova GmbH

2.5.5 AuthenticEvent.clientX

Siehe auch

Deklaration: clientX als long

Beschreibung
X-Wert der aktuellen Mausposition in den Client-Koordinaten.

2.5.6 AuthenticEvent.clientY

Siehe auch

Deklaration: clientY als long

Beschreibung
Y-Wert der aktuellen Mausposition in den Client-Koordinaten.

© 2010 Altova GmbH

2.5.7 AuthenticEvent.ctrlKey

Siehe auch

Deklaration: ctrlKey als Boolean

Beschreibung
True, wenn die rechte Strg-Taste gedrückt wird.

2.5.8 AuthenticEvent.ctrlLeft

Siehe auch

Deklaration: ctrlLeft als Boolean

Beschreibung
True, wenn die linke Strg-Taste gedrückt wird.

© 2010 Altova GmbH

2.5.9 AuthenticEvent.dataTransfer

Siehe auch

Deklaration: dataTransfer als Variant

Beschreibung
Eigenschaft dataTransfer

2.5.10 AuthenticEvent.fromElement

Siehe auch

Deklaration: fromElement als Variant

Beschreibung
Derzeit wird diese Eigenschaft von keinem Ereignis gesetzt.

© 2010 Altova GmbH

2.5.11 AuthenticEvent.keyCode

Siehe auch

Deklaration: keyCode als long

Beschreibung
Tastaturcode der aktuell gedrückten Taste.

Diese Eigenschaft ist mit Lese- und Schreibzugriff.

2.5.12 AuthenticEvent.propertyName

Siehe auch

Deklaration: propertyName als String

Beschreibung
Derzeit wird diese Eigenschaft von keinem Ereignis gesetzt.

© 2010 Altova GmbH

2.5.13 AuthenticEvent.repeat

Siehe auch

Deklaration: repeat als Boolean

Beschreibung
True, wenn sich das onkeydown-Ereignis wiederholt.

2.5.14 AuthenticEvent.returnValue

Siehe auch

Deklaration: returnValue als Variant

Beschreibung
Verwenden Sie returnValue, um einen Rückgabewert für Ihren Ereignishandler zu setzen.

© 2010 Altova GmbH

2.5.15 AuthenticEvent.shiftKey

Siehe auch

Deklaration: shiftKey als Boolean

Beschreibung
True, wenn die rechte Umschalttaste gedrückt wird.

2.5.16 AuthenticEvent.shiftLeft

Siehe auch

Deklaration: shiftLeft als Boolean

Beschreibung
True, wenn die linke Umschalttaste gedrückt wird.

© 2010 Altova GmbH

2.5.17 AuthenticEvent.srcElement

Siehe auch

Deklaration: srcElement als Variant

Beschreibung
Element, das das aktuelle Ereignis auslöst.
Dies ist normalerweise ein XMLData-Objekt.

Seit Version 3.0.0.0:

Die Eigenschaft kann auch eine Referenz auf ein AuthenticCommand-Objekt enthalten, wenn Sie von einem ondoceditcommand-Ereignis aus gesetzt wurde.

2.5.18 AuthenticEvent.type

Siehe auch

Deklaration: type als String

Beschreibung
Derzeit wird diese Eigenschaft von keinem Ereignis gesetzt.

© 2010 Altova GmbH

2.6 AuthenticLoadObject

Siehe auch

Eigenschaften
String
URL

Beschreibung
Das XMLSpyXMLLoadSave-Objekt dient dazu, die Quelle für die Dateien festzulegen, die geladen werden müssen. Sie können den Inhalt entweder direkt über die String-Eigenschaft festlegen oder als externen Pfad über die URL-Eigenschaft.

Nähere Informationen zur Verwendung der Eigenschaften finden Sie auch unter Authentic.SchemaLoadObject, Authentic.DesignDataLoadObject und Authentic.XMLDataLoadObject.

2.6.1 AuthenticLoadObject.String

Siehe auch

Deklaration: String als String

Beschreibung
Mit Hilfe dieser Eigenschaft können Sie die XML-Struktur von einem String aus setzen. Wenn Sie diese Eigenschaft verwenden möchten, muss die URL-Eigenschaft des Objekts leer sein.

© 2010 Altova GmbH

2.6.2 AuthenticLoadObject.URL

Siehe auch

Deklaration: URL als String

Beschreibung
Die Eigenschaft sollte eine gültige URL für den Lade- oder Speichervorgang enthalten. Derzeit werden die HTTP-Protokolle http, https, ftp und gopher unterstützt.

2.7 AuthenticRange

Siehe auch

In der ersten Tabelle werden die Eigenschaften und Methoden von AuthenticRange aufgelistet, mit Hilfe derer Sie durch das Dokument navigieren und bestimmte Abschnitte auswählen können.

Eigenschaften		Methoden
Application	Clone	MoveBegin
FirstTextPosition	CollapsToBegin	MoveEnd
FirstXMLData	CollapsToEnd	NextCursorPosition
FirstXMLDataOffset	ExpandTo	PreviousCursorPosition
LastTextPosition	Goto	Select
LastXMLData	GotoNext	SelectNext
LastXMLDataOffset	GotoPrevious	SelectPrevious
Parent	IsEmpty	SetFromRange
	IsEqual	

In der folgenden Tabelle, werden die Methoden zum Ändern des Inhalts aufgelistet. Viele davon können durch Rechtsklick mit der Maus aufgerufen werden.

Eigenschaften	Bearbeitungsoperationen	Operationen für dynamische Tabellen
Text	Copy	AppendRow
	Cut	DeleteRow
	Delete	DuplicateRow
	Paste	InsertRow
		IsInDynamicTable
		MoveRowDown
		MoveRowUp

Die folgenden Methoden stellen die Funktionalitäten der Authentic-Eingabehilfenfenster für Range (Bereichs)-Objekte bereit.

Operationen der Eingabehilfenfenster

Elemente	Attribute	Entities
CanPerformActionWith	GetElementAttributeValue	GetEntityNames
CanPerformAction	GetElementAttributeNames	InsertEntity
PerformAction	GetElementHierarchy	
	HasElementAttribute	
	SetElementAttributeValue	

Beschreibung
Bei AuthenticRange-Objekten der Automation-Schnittstelle handelt es sich um die Selektion. Sie können damit den Cursor auf jede Cursorposition in der Authentic-Ansicht setzen oder einen Abschnitt des Dokuments auswählen. Die für AuthenticRange-Objekte verfügbaren Operationen haben dann auf diese Auswahl dieselbe Wirkung wie die entsprechenden Operationen der Benutzeroberfläche auf die aktuelle Auswahl in der Authentic-Ansicht. Der Hauptunterschied ist, dass Sie eine beliebige Anzahl von AuthenticRange-Objekten gleichzeitig verwenden können, während es bei der Benutzeroberfläche nur genau eine Selektion gibt.

Um zunächst einmal ein Bereichsobjekt zu erhalten, verwenden Sie AuthenticView.Selection, um einen Bereich zu erhalten, der der aktuellen Selektion auf der Benutzeroberfläche

© 2010 Altova GmbH

entspricht. Alternativ dazu können einige einfache Bereiche auch über die schreibgeschützten Eigenschaften AuthenticView.DocumentBegin, AuthenticView.DocumentEnd und AuthenticView.WholeDocument aufgerufen werden. Die flexibelste Methode ist AuthenticView.Goto, mit Hilfe derer Sie in nur einem Aufruf zu einem bestimmten Bereich im Dokument navigieren können. Für komplexere Auswahlvorgänge können Sie die obigen Methoden kombinieren. Die verschiedenen Navigationsmethoden zu Bereichsobjekten sind in der ersten Tabelle auf dieser Seite aufgelistet.

Ein andere Methode, einen Dokumentenabschnitt auszuwählen, ist die Verwendung der Positionseigenschaften des Bereichsobjekts. Es stehen zwei Systeme zur Positionierung zur Verfügung, die beliebig kombiniert werden können:

- **Absolute** Textcursor-Positionen, beginnend mit Position 0 am Dokumentanfang, können für den Anfang und das Ende eines Bereichs gesetzt und abgerufen werden. Nähere Informationen dazu finden Sie unter FirstTextPosition und LastTextPosition. Zur Verwendung dieser Methode sind komplexe interne Berechnungen erforderlich, daher sollte sie mit Vorsicht verwendet werden.
- Das **XMLData-**Element und eine Textposition innerhalb dieses Elements können für den Anfang und das Ende eines Bereichs gesetzt und abgerufen werden. Nähere Informationen dazu finden Sie unter FirstXMLData, FirstXMLDataOffset, LastXMLData und LastXMLDataOffset. Diese Methode ist äußerst effizient, erfordert aber Kenntnisse über die zugrunde liegende Dokumentenstruktur. Sie können damit XMLData-Objekte suchen und Operationen daran durchführen, die sonst über die Benutzeroberfläche nicht zur Verfügung stehen.

Änderungen am Dokumenteninhalt können durch verschiedene Methoden vorgenommen werden:

- mit Hilfe der Text-Eigenschaft können Sie den durch das Bereichsobjekt ausgewählten Dokumententext abrufen. Ist sie gesetzt, wird der ausgewählte Dokumententext durch den neuen Text ersetzt.
- die Standard-Dokumentbearbeitungsfunktionen Cut, Copy, Paste und Delete.
- Tabellenoperationen für Tabellen, die dynamisch wachsen können.
- Methoden, die die Funktionalitäten der Authentic-Eingabehilfenfenster abbilden.
- Zugriff auf die XMLData-Objekte des zugrunde liegenden Dokuments, damit diese direkt geändert werden können.

2.7.1 AuthenticRange.AppendRow

Siehe auch

Methode: AppendRow () als Boolean

Beschreibung
Wenn sich der Anfang des Bereichs innerhalb einer dynamischen Tabelle befindet, wird mit der Methode eine neue Zeile am Ende der ausgewählten Tabelle eingefügt. Die Auswahl des Bereichs wird geändert, sodass sie auf den Beginn der neuen Zeile zeigt. Die Funktion gibt den Wert *true* zurück, wenn das Anhängen erfolgreich war. Andernfalls wird *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------------------
'                        VBScript
' Append row at end of current dynamically growable table
' ------------------------------------------------------
Dim objRange
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection

' check if we can insert something
If objRange.IsInDynamicTable Then
     objRange.AppendRow
     ' objRange points to beginning of new row
     objRange.Select
End If
```


© 2010 Altova GmbH

2.7.2 AuthenticRange.Application

Siehe auch

Eigenschaft: Application als Authentic (read-only)

Beschreibung
Liefert das Applikationsobjekt.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.3 AuthenticRange.CanPerformAction

Siehe auch

Methode: CanPerformAction (*eAction* als SPYAuthenticActions, *strElementName* als String) als Boolean

Beschreibung
CanPerformAction und die dazugehörigen Methoden gestatten den Zugriff auf die Eingabehilfenfunktionen von Authentic. Mit Hilfe dieser Funktion können Sie den Inhalt eines Dokuments einfach und auf konsistente Weise bearbeiten, ohne wissen zu müssen, wo genau die Änderung vorgenommen wird. Der Anfang des Bereichsobjekts dient dazu, die nächste gültige Position zu finden, an der die angegebene Aktion durchgeführt werden kann. Wird diese Position gefunden, gibt die Methode *True* zurück, andernfalls *False*.

HINWEIS: Um alle gültigen Elementnamen für eine bestimmte Aktion zu finden, verwenden Sie CanPerformActionWith.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2007	Es wurde eine ungültige Aktion definiert.

Beispiele
Siehe PerformAction.

© 2010 Altova GmbH

2.7.4 AuthenticRange.CanPerformActionWith

Siehe auch

Methode: CanPerformActionWith (*eAction* als SPYAuthenticActions, *out_arrElementNames* als Variant)

Beschreibung
CanPerformActionWith und die dazugehörigen Methoden ermöglichen den Zugriff auf die Eingabehilfenfunktionen von Authentic. Mit Hilfe dieser Funktion können Sie den Inhalt eines Dokuments einfach und auf konsistente Weise bearbeiten, ohne wissen zu müssen, wo genau die Änderung vorgenommen wird.

Diese Methode gibt einen Array der Elementnamen zurück, an denen die angegebene Aktion ausgeführt werden kann.

HINWEIS: Verwenden Sie PerformActionWith, um die Aktion anzuwenden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2007	Es wurde eine ungültige Aktion definiert.

Beispiele
Siehe PerformAction.

2.7.5 AuthenticRange.Clone

Siehe auch

Methode: Clone () als AuthenticRange

Beschreibung
Gibt eine Kopie des Bereichsobjekts zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.6 AuthenticRange.CollapsToBegin

Siehe auch

Methode: CollapsToBegin () als AuthenticRange

Beschreibung
Setzt das Ende des Bereichsobjekts and seinen Beginn. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.7 AuthenticRange.CollapsToEnd

Siehe auch

Methode: CollapsToEnd () als AuthenticRange

Beschreibung
Setzt den Beginn des Bereichsobjekts an seinen Anfang. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.8 AuthenticRange.Copy

Siehe auch

Methode: Copy () als Boolean

Beschreibung
Gibt *False* zurück, wenn der Bereich keine Dokumentabschnitte enthält, die kopiert werden können.
Gibt *True* zurück, wenn Text sowie - im Falle von vollständig selektierten XML-Elementen - die Elemente selbst in den Kopier-/Einfügepuffer kopiert wurden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.9 AuthenticRange.Cut

Siehe auch

Methode: Cut () als Boolean

Beschreibung
Gibt *False* zurück, wenn der Bereich Dokumentabschnitte enthält, die nicht gelöscht werden dürfen.
Gibt den Wert *True* zurück, wenn Text sowie - im Falle von vollständig selektierten XML-Elementen - die Elemente selbst aus dem Dokument gelöscht wurden und im Kopier-/Einfügepuffer gespeichert wurden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.10 AuthenticRange.Delete

Siehe auch

Methode: Delete () als Boolean

Beschreibung
Gibt *False* zurück, wenn der Bereich Dokumentabschnitte enthält, die nicht gelöscht werden dürfen.
Gibt den Wert *True* zurück, wenn Text sowie - im Falle von vollständig selektierten XML-Elementen - die Elemente selbst aus dem Dokument gelöscht wurden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.11 AuthenticRange.DeleteRow

Siehe auch

Methode: DeleteRow () als Boolean

Beschreibung
Wenn sich der Anfang des Bereichs innerhalb einer dynamischen Tabelle befindet, löscht diese Methode die ausgewählte Zeile. Die Auswahl des Bereichs wird geändert und zeigt auf das nächste Element nach der gelöschten Zeile. Die Funktion gibt *true* zurück, wenn der Löschvorgang erfolgreich war, andernfalls wird *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------------------
'                      VBScript
' Delete selected row from dynamically growing table
' ------------------------------------------------------
Dim objRange
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection

' check if we are in a table
If objRange.IsInDynamicTable Then
     objRange.DeleteRow
End If
```


© 2010 Altova GmbH

2.7.12 AuthenticRange.DuplicateRow

Siehe auch

Methode: DuplicateRow () als Boolean

Beschreibung
Wenn sich der Anfang des Bereichs innerhalb einer dynamischen Tabelle befindet, fügt diese Methode eine Kopie der aktuellen Zeile hinter der ausgewählten Zeile ein. Die Funktion gibt *true* zurück, wenn der Kopiervorgang erfolgreich war, andernfalls wird *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------------------
'                       VBScript
' dupplicate row in current dynamically growable table
' ------------------------------------------------------
Dim objRange
Set objRange = objPlugin.ActiveDocument.AuthenticView.Selection

' check if we can insert soemthing
If objRange.IsInDynamicTable Then
     objRange.DuplicateRow
     ' objRange points to begining of new row
     objRange.Select
End If
```

2.7.13 AuthenticRange.ExpandTo

Siehe auch

Methode: ExpandTo (*eKind* als SPYAuthenticElementKind) als AuthenticRange

Beschreibung
Wählt das gesamte Element vom Typ *eKind* aus, das an der ersten Cursorposition des Bereichs beginnt bzw. diese enthält. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Die Erweiterung des Bereichs würde über das Ende des Dokuments hinausgehen.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.14 AuthenticRange.FirstTextPosition

Siehe auch

Eigenschaft: FirstTextPosition als Long

Beschreibung
Setzt oder holt den am weitesten links befindlichen Textpositionsindex des Bereichsobjekts. Dieser Index ist immer kleiner oder gleich der LastTextPosition. Die Indizierung beginnt am Anfang des Dokuments mit 0 und erhöht sich schrittweise mit jeder neuen Position, an der sich der Textcursor befinden kann. Eine Erhöhung der Textposition um 1 hat dieselbe Wirkung wie die Rechtspfeil-Taste. Eine Verringerung der Textposition um den Wert 1 hat dieselbe Wirkung wie die Linkspfeil-Taste.

Wenn Sie *FirstTextPosition* auf einen Wert setzen, der größer ist als die aktuelle LastTextPosition, wird LastTextPosition auf den Wert der neuen *FirstTextPosition* gesetzt.

HINWEIS: Verwenden Sie Textcursorpositionen mit Vorsicht, da dies im Vergleich zur Cursorpositionierung auf Basis von XMLData ein aufwändiger Vorgang ist.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2006	Es wurde eine Textposition außerhalb des Dokuments angegeben

Beispiele

```
' ----------------------------------------
'               VBScript
' ----------------------------------------
Dim objAuthenticView
' we assume that the active document is open in authentic view mode
Set objAuthenticView = objPlugin.AuthenticView

nDocStartPosition = objAuthenticView.DocumentBegin.FirstTextPosition
nDocEndPosition = objAuthenticView.DocumentEnd.FirstTextPosition

' let's create a range that selects the whole document
' in an inefficient way
Dim objRange
' we need to get a (any) range object first
Set objRange = objAuthenticView.DocumentBegin
objRange.FirstTextPosition = nDocStartPosition
objRange.LastTextPosition = nDocEndPosition

' let's check if we got it right
If objRange.isEqual(objAuthenticView.WholeDocument) Then
     MsgBox "Test using direct text cursor positioning was ok"
Else
     MsgBox "Ooops!"
End If
```

2.7.15 AuthenticRange.FirstXMLData

Siehe auch

Eigenschaft: FirstXMLData als XMLData

Beschreibung
Setzt oder holt das erste XMLData-Element im zugrunde liegenden Dokument, das zum Teil oder zur Gänze vom Bereich ausgewählt ist. Der genaue Beginn der Auswahl wird durch das FirstXMLDataOffset Attribut definiert.

Wenn FirstXMLData auf ein neues Datenobjekt gesetzt wird, wird FirstXMLDataOffset auf die erste Cursorposition innerhalb dieses Elements gesetzt. Es dürfen nur XMLData-Objekte verwendet werden, für die eine Cursorposition möglich ist. Wenn Sie *FirstXMLData* setzen / wählt FirstXMLDataOffset eine Position aus, die größer ist als der aktuelle Wert von LastXMLData / LastXMLDataOffset. Das zweite wird auf die neue Anfangsposition verschoben.

HINWEIS: Sie können mit Hilfe der Eigenschaften FirstXMLData und LastXMLData direkt auf das zugrunde liegende XML-Dokument zugreifen und dieses manipulieren, wenn die mit dem AuthenticRange-Objekt verfügbaren Methoden nicht ausreichen.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2008	Interner Fehler
2009	Das XMLData-Objekt kann nicht aufgerufen werden.

Beispiele

```
' ------------------------------------------------
'                 VBScript
' show name of currently selected XMLData element
' ------------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

Dim objXmlData
Set objXMLData = objAuthenticView.Selection.FirstXMLData
' authentic view adds a 'text' child element to elements
' of the document which have content. So we have to go one
' element up.
Set objXMLData = objXMLData.Parent
MsgBox "Current selection selects element " & objXMLData.Name
```


© 2010 Altova GmbH

2.7.16 AuthenticRange.FirstXMLDataOffset

Siehe auch

Eigenschaft: FirstXMLDataOffset als Long

Beschreibung
Setzt oder holt den Cursorpositions-Offset im FirstXMLData-Element für den Anfang des Bereichs. Die Offset-Positionen basieren auf den von der Text-Eigenschaft zurückgegebenen Zeichen und beginnen mit dem Wert 0. Verwenden Sie beim Setzen eines neuen Offset den Wert -1, um den Offset an die letzte mögliche Position im Element zu setzen. Seien Sie besonders in den folgenden Fällen vorsichtig:

- Die Textform von Einträgen in Auswahllisten, Kontrollkästchen und ähnlichen Steuerelementen unterscheidet sich unter Umständen von dem, was auf dem Bildschirm angezeigt wird. Der Daten-Offset basiert zwar auf diesem Text, doch gibt es nur zwei gültige Offset-Positionen, eine am Anfang und eine am Ende des Eintrags. Wenn Sie versuchen, den Offset in die Mitte des Eintrags zu positionieren, wird der Offset an das Ende des Eintrags gesetzt.
- Die Textform von XML Entities kann sich in ihrer Darstellung von der auf dem Bildschirm dargestellten Länge unterscheiden. Der Offset basiert auf der Textform der Entities.

Wenn *FirstXMLData* / FirstXMLDataOffset eine Position hinter dem aktuellen LastXMLData / LastXMLDataOffset auswählt, wird letzteres an die neue Anfangsposition verschoben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Offset-Wert angegeben. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'               VBScript
' Select the complete text of an XMLData element
' using XMLData based selection and ExpandTo
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

' first we use the XMLData based range properties
' to select all text of the first XMLData element
' in the current selection
Dim objRange
Set objRange = objAuthenticView.Selection
objRange.FirstXMLDataOffset = 0  ' start at beginning of element text
objRange.LastXMLData = objRange.FirstXMLData  ' select only one element
objRange.LastXMLDataOffset = -N  ' select till its end

' the same can be achieved with the ExpandTo method
Dim objRange2
Set objRange2 = objAuthenticView.Selection.ExpandTo(spyAuthenticTag)

' were we successful?
If objRange.IsEqual(objRange2) Then
     objRange.Select()
Else
     MsgBox "Ooops"
```

```
End If
```


© 2010 Altova GmbH

2.7.17 AuthenticRange.GetElementAttributeNames

Siehe auch

Methode: GetElementAttributeNames (*strElementName* als String, *out_arrAttributeNames* als Variant)

Beschreibung
Liefert die Namen aller Attribute für das einschließende Element mit dem angegebene Namen. Verwenden Sie die Element- / Attribut-Paare, um den Attributwert mit Hilfe der Methoden GetElementAttributeValue und SetElementAttributeValue zu setzen oder zu holen.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Elementname angegeben. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele
Siehe SetElementAttributeValue.

2.7.18 AuthenticRange.GetElementAttributeValue

Siehe auch

Methode: GetElementAttributeValue (*strElementName* als String, *strAttributeName* als String) als String

Beschreibung
Liefert für das mit *strElementName* definierte Element den Wert des in *strAttributeName* definierten Attributs. Wenn das Attribut unterstützt wird, ihm jedoch kein Wert zugewiesen wurde, wird ein leerer String zurückgegeben. Um herauszufinden, welche Attribute von einem Element unterstützt werden, verwenden Sie GetElementAttributeNames oder HasElementAttribute.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Elementname angegeben. Es wurde ein ungültiger Attributname angegeben. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele
Siehe SetElementAttributeValue.

© 2010 Altova GmbH

2.7.19 AuthenticRange.GetElementHierarchy

Siehe auch

Methode: GetElementHierarchy (*out_arrElementNames* als Variant)

Beschreibung
Liefert die Namen aller XML-Elemente, die Parent-Elemente der aktuellen Auswahl sind. Innere Elemente werden vor den diese einschließenden Elementen aufgelistet. Wenn sich die aktuelle Auswahl nicht innerhalb eines einzelnen XMLData-Elements befindet, wird eine leere Liste zurückgegeben.

Zusammen mit dem Bereichsobjekt identifizieren die Namen der Elementhierarchie XMLData-Elemente eindeutig. Die Attribute dieser Elemente können direkt durch GetElementAttributeNames und die dazugehörigen Methoden aufgerufen werden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele
Siehe SetElementAttributeValue.

2.7.20 AuthenticRange.GetEntityNames

Siehe auch

Methode: GetEntityNames (*out_arrEntityNames* als Variant)

Beschreibung
Liefert die Namen aller definierten Entities. Die Liste der abgerufenen Entities ist unabhängig von der aktuellen Auswahl oder der aktuellen Position. Verwenden Sie einen dieser Namen mit der InsertEntity-Funktion.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele
Siehe InsertEntity.

© 2010 Altova GmbH

2.7.21 AuthenticRange.Goto

Siehe auch

Methode: Goto (*eKind* als SPYAuthenticElementKind, *nCount* als Long, *eFrom* als SPYAuthenticDocumentPosition) als AuthenticRange

Beschreibung
Setzt den Bereich an den Anfang des *nCount*-Elements vom Typ *eKind*. Die Anfangsposition wird durch den Parameter *eFrom* definiert.

Verwenden Sie zum Navigieren zum Ende des Dokuments positive Positionswerte für *nCount.* Verwenden Sie negative Werte, um zum Anfang des Dokuments zu navigieren. Die Methode gibt das geänderte Range-Objekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2004	Das Ziel befindet sich vor dem Anfang des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Es wurde eine ungültige Anfangsposition definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.22 AuthenticRange.GotoNext

Siehe auch

Methode: GotoNext (*eKind* als SPYAuthenticElementKind) als AuthenticRange

Beschreibung
Setzt den Bereich an den Anfang des nächsten Elements vom Typ *eKind*. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'                VBScript
' Scan through the whole document word-by-word
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

Dim objRange
Set objRange = objAuthenticView.DocumentBegin
Dim bEndOfDocument
bEndOfDocument = False

On Error Resume Next
While Not bEndOfDocument
     objRange.GotoNext(spyAuthenticWord).Select
     If ((Err.number - vbObjecterror) = 2003) Then
            bEndOfDocument = True
            Err.Clear
     ElseIf (Err.number <> 0) Then
            Err.Raise  ' forward error
     End If
Wend
```


© 2010 Altova GmbH

2.7.23 AuthenticRange.GotoNextCursorPosition

Siehe auch

Methode: GotoNextCursorPosition () als AuthenticRange

Beschreibung
Setzt den Bereich auf die nächste Cursorposition hinter dem Ende der aktuellen Endposition. Gibt das geänderte Objekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.24 AuthenticRange.GotoPrevious

Siehe auch

Methode: GotoPrevious (*eKind* als SPYAuthenticElementKind) als AuthenticRange

Beschreibung
Setzt den Bereich an den Anfang des Elements vom Typ *eKind*, welches sich vor dem Anfang des aktuellen Bereichs befindet. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2004	Das Ziel befindet sich vor dem Anfang des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'                  VBScript
' Scan through the whole document tag-by-tag
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

Dim objRange
Set objRange = objAuthenticView.DocumentEnd
Dim bEndOfDocument
bBeginOfDocument = False

On Error Resume Next
While Not bBeginOfDocument
     objRange.GotoPrevious(spyAuthenticTag).Select
     If ((Err.number - vbObjecterror) = 2004) Then
            bBeginOfDocument = True
            Err.Clear
     ElseIf (Err.number <> 0) Then
            Err.Raise  ' forward error
     End If
Wend
```


© 2010 Altova GmbH

2.7.25 AuthenticRange.GotoPreviousCursorPosition

Siehe auch

Methode: GotoPreviousCursorPosition () als AuthenticRange

Beschreibung
Setzt den Bereich der Cursorposition unmittelbar vor die aktuelle Position. Gibt das geänderte Objekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2004	Das Ziel befindet sich vor dem Anfang des Dokuments.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.26 AuthenticRange.HasElementAttribute

Siehe auch

Methode: HasElementAttribute (*strElementName* als String, *strAttributeName* als String) als Boolean

Beschreibung
Testet, ob das einschließende Element mit dem Namen *strElementName* das in *strAttributeName* definierte Attribut unterstützt.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Elementname definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.27 AuthenticRange.InsertEntity

Siehe auch

Methode: InsertEntity (*strEntityName* als String)

Beschreibung
Ersetzt die Bereichsauswahl durch die angegebene Entity. Die Entity muss einer der Entity-Namen sein, die von GetEntityNames zurückgegeben werden.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Name für den Eintrag angegeben.

Beispiele

```
' ------------------------------------------------------
'                          VBScript
' Insert the first entity in the list of availabel entities
' ------------------------------------------------------
Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' first we get the names of all available entities as they
' are shown in the entry helper of XMLSpy
Dim arrEntities
objRange.GetEntityNames arrEntities

' we insert the first one of the list
If UBound(arrEntities) >= 0 Then
     objRange.InsertEntity arrEntities(0)
     objRange.Select()
Else
     MsgBox "Sorry, no entities are available for this document"
End If
```

2.7.28 AuthenticRange.InsertRow

Siehe auch

Methode: InsertRow () als Boolean

Beschreibung
Wenn sich der Anfang des Bereichs innerhalb einer dynamischen Tabelle befindet, wird mit der Methode eine neue Zeile vor der aktuellen eingefügt. Die Auswahl des Bereichs wird geändert, sodass sie auf den Beginn der neu eingefügten Zeile zeigt. Die Funktion gibt den Wert *true* zurück, wenn das Einfügen erfolgreich war. Andernfalls wird *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------------------------
'                       VBScript
' Insert row at beginning of current dynamically growing table
' ------------------------------------------------------------
Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' check if we can insert something
If objRange.IsInDynamicTable Then
     objRange.InsertRow
     ' objRange points to beginning of new row
     objRange.Select
End If
```


© 2010 Altova GmbH

2.7.29 AuthenticRange.IsEmpty

Siehe auch

Methode: IsEmpty () als Boolean

Beschreibung
Testet, ob die erste und die letzte Position des Bereichs identisch sind.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.30 AuthenticRange.IsEqual

Siehe auch

Methode: IsEqual (*objCmpRange* als AuthenticRange) als Boolean

Beschreibung
Testet, ob der Beginn und das Ende der beiden Bereiche identisch sind.

Fehler

2001	Eines der beiden verglichenen Bereichsobjekte ist ungültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.31 AuthenticRange.IsInDynamicTable

Siehe auch

Methode: IsInDynamicTable () als Boolean

Beschreibung
Testet, ob sich der Anfang eines Bereichs innerhalb einer Tabelle befindet, die andere Zeilenoperationen unterstützt.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.32 AuthenticRange.LastTextPosition

Siehe auch

Eigenschaft: LastTextPosition als Long

Beschreibung
Setzt oder holt den am weitesten rechts befindlichen Textpositionsindex des Bereichsobjekts. Dieser Index ist immer größer oder gleich der FirstTextPosition. Die Indizierung beginnt am Anfang des Dokuments mit 0 und erhöht sich schrittweise mit jeder neuen Position, an der sich der Textcursor befinden kann. Eine Erhöhung der Textposition um 1 hat dieselbe Wirkung wie die Rechtspfeil-Taste. Eine Verringerung der Textposition um den Wert 1 hat dieselbe Wirkung wie die Linkspfeil-Taste.

Wenn Sie *LastTextPosition* auf einen Wert setzen, der kleiner ist als die aktuelle FirstTextPosition, wird FirstTextPosition auf den Wert der neuen *LastTextPosition* gesetzt.

HINWEIS: Verwenden Sie Textcursorpositionen mit Vorsicht, da dies im Vergleich zur Cursorpositionierung auf Basis von XMLData ein aufwändiger Vorgang ist.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2006	Es wurde eine Textposition außerhalb des Dokuments angegeben

Beispiele

```
' ---------------------------------------
'                VBScript
' ---------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

nDocStartPosition = objAuthenticView.DocumentBegin.FirstTextPosition
nDocEndPosition = objAuthenticView.DocumentEnd.FirstTextPosition

' let's create a range that selects the whole document
' in an inefficient way
Dim objRange
' we need to get a (any) range object first
Set objRange = objAuthenticView.DocumentBegin
objRange.FirstTextPosition = nDocStartPosition
objRange.LastTextPosition = nDocEndPosition

' let's check if we got it right
If objRange.isEqual(objAuthenticView.WholeDocument) Then
     MsgBox "Test using direct text cursor positioning was ok"
Else
     MsgBox "Ooops!"
End If
```


© 2010 Altova GmbH

2.7.33 AuthenticRange.LastXMLData

Siehe auch

Eigenschaft: LastXMLData als XMLData

Beschreibung
Setzt oder holt das letzte XMLData-Element im zugrunde liegenden Dokument, das zum Teil oder zur Gänze vom Bereich ausgewählt ist. Das genaue Ende der Auswahl wird durch das LastXMLDataOffset Attribut definiert.

Wenn *LastXMLData* auf ein neues Datenobjekt gesetzt wird, wird LastXMLDataOffset auf die letzte Cursorposition innerhalb dieses Elements gesetzt. Es dürfen nur XMLData-Objekte verwendet werden, für die es eine Cursorposition gibt. Wenn Sie *LastXMLData* / LastXMLDataOffset setzen, wählen Sie eine Position aus, die kleiner ist als der aktuelle Wert von FirstXMLData / FirstXMLDataOffset. Letzteres wird auf die neue Endposition verschoben.

HINWEIS: Sie können mit Hilfe der Eigenschaften FirstXMLData und LastXMLData direkt auf das zugrunde liegende XML-Dokument zugreifen und dieses manipulieren, wenn die mit dem AuthenticRange-Objekt verfügbaren Methoden nicht ausreichen.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2008	Interner Fehler
2009	Das XMLData-Objekt kann nicht aufgerufen werden.

2.7.34 AuthenticRange.LastXMLDataOffset

Siehe auch

Eigenschaft: LastXMLDataOffset als Long

Beschreibung
Setzt oder holt die Cursorposition im LastXMLData-Element für das Ende des Bereichs.

Die Offset-Positionen basieren auf den von der Text-Eigenschaft zurückgegebenen Zeichen und beginnen mit dem Wert 0. Verwenden Sie beim Setzen eines neuen Offset den Wert -1, um den Offset an die letzte mögliche Position im Element zu setzen. Seien Sie besonders in den folgenden Fällen vorsichtig:

- Die Textform von Einträgen in Auswahllisten, Kontrollkästchen und ähnlichen Steuerelementen unterscheidet sich unter Umständen von dem, was auf dem Bildschirm angezeigt wird. Der Daten-Offset basiert zwar auf diesem Text, doch gibt es nur zwei gültige Offset-Positionen, eine am Anfang und eine am Ende des Eintrags. Wenn Sie versuchen, den Offset in die Mitte des Eintrags zu positionieren, wird der Offset an das Ende des Eintrags gesetzt.
- Die Textform von XML Entities kann sich in ihrer Darstellung von der auf dem Bildschirm dargestellten Länge unterscheiden. Der Offset basiert auf der Textform der Entities z.B. &.

Wenn LastXMLData / LastXMLDataOffset eine Position vor dem aktuellen FirstXMLData / FirstXMLDataOffset auswählt, wird letzteres an die neue Endposition verschoben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Offset-Wert angegeben. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'                 VBScript
' Select the complete text of an XMLData element
' using XMLData based selection and ExpandTo
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

' first we use the XMLData based range properties
' to select all text of the first XMLData element
' in the current selection
Dim objRange
Set objRange = objAuthenticView.Selection
objRange.FirstXMLDataOffset = 0  ' start at beginning of element text
objRange.LastXMLData = objRange.FirstXMLData  ' select only one element
objRange.LastXMLDataOffset = -N  ' select till its end

' the same can be achieved with the ExpandTo method
Dim objRange2
Set objRange2 = objAuthenticView.Selection.ExpandTo(spyAuthenticTag)

' were we successful?
If objRange.IsEqual(objRange2) Then
     objRange.Select()
Else
```


© 2010 Altova GmbH

```
        MsgBox "Ooops"
End If
```

2.7.35 AuthenticRange.MoveBegin

Siehe auch

Methode: MoveBegin (*eKind* als SPYAuthenticElementKind, *nCount* als Long) als AuthenticRange

Beschreibung
Verschiebt den Anfang des Bereichs an den Anfang des *nCount* Elements vom Typ *eKind*. Die Zählung beginnt am aktuellen Beginn des Bereichsobjekts.

Mit Hilfe von positiven Zahlen für *nCount* können Sie die Position zum Ende des Dokuments hin verschieben, mit Hilfe von negativen Zahlen können Sie die Position zum Beginn des Dokuments hin verschieben. Das Ende des Bereichs wird nicht verschoben, es sei denn, der neue Anfang des Bereichs wäre größer als der Wert. In diesem Fall wird das Ende auf den neuen Anfang verschoben. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2004	Das Ziel befindet sich vor dem Anfang des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.36 AuthenticRange.MoveEnd

Siehe auch

Methode: MoveEnd (*eKind* als SPYAuthenticElementKind, *nCount* als Long) als AuthenticRange

Beschreibung
Verschiebt das Ende des Bereichs an den Anfang des *nCount* Elements vom Typ *eKind*. Die Zählung beginnt am aktuellen Beginn des Bereichsobjekts.

Mit Hilfe von positiven Zahlen für *nCount* können Sie die Position zum Ende des Dokuments hin verschieben, mit Hilfe von negativen Zahlen können Sie die Position zum Beginn des Dokuments hin verschieben. Das Ende des Bereichs wird nicht verschoben, es sei denn, der neue Anfang des Bereichs wäre größer als der Wert. In diesem Fall wird das Ende an den neuen Anfang verschoben. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2004	Das Ziel befindet sich vor dem Anfang des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.37 AuthenticRange.MoveRowDown

Siehe auch

Methode: MoveRowDown () als Boolean

Beschreibung
Wenn sich der Anfang eines Bereichs innerhalb einer dynamischen Tabelle befindet und eine Zeile ausgewählt wird, die nicht die letzte Zeile in dieser Tabelle ist, vertauscht diese Methode diese Zeile mit der Zeile unmittelbar danach. Die Auswahl des Bereichs verschiebt sich mit der Zeile, ändert sich aber sonst nicht. Die Funktion gibt den Wert *true* zurück, wenn das Verschieben erfolgreich war, ansonsten wird der Wert *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.38 AuthenticRange.MoveRowUp

Siehe auch

Methode: MoveRowUp () als Boolean

Beschreibung
Wenn sich der Anfang eines Bereichs innerhalb einer dynamischen Tabelle befindet und eine Zeile ausgewählt wird, die nicht die erste Zeile in dieser Tabelle ist, vertauscht diese Methode diese Zeile mit der Zeile unmittelbar oberhalb davon. Die Auswahl des Bereichs verschiebt sich mit der Zeile, ändert sich aber sonst nicht. Die Funktion gibt den Wert *true* zurück, wenn das Verschieben erfolgreich war, ansonsten wird der Wert *false* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele
Siehe JScript - Bubble Sort Dynamic Tables.

2.7.39 AuthenticRange.Parent

Siehe auch

Eigenschaft: Parent als AuthenticView (read-only)

Beschreibung
Ruft die Ansicht auf, die dieses Bereichsobjekt enthält.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.7.40 AuthenticRange.Paste

Siehe auch

Methode: Paste () als Boolean

Beschreibung
Gibt *False* zurück, wenn der Kopier-/Einfügepuffer leer ist oder die aktuelle Auswahl nicht durch den Inhalt dieses Puffers ersetzt werden kann.

Andernfalls wird die aktuelle Auswahl gelöscht, der Inhalt dieses Puffers eingefügt und der Wert *true* zurückgegeben.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.7.41 AuthenticRange.PerformAction

Siehe auch

Methode: PerformAction (*eAction* als SPYAuthenticActions, *strElementName* als String) als Boolean

Beschreibung
PerformAction und die dazugehörigen Methoden verleihen Zugriff auf die Eingabehilfenfunktionen von Authentic. Mit Hilfe dieser Funktion können Sie Dokumentinhalt schnell und auf konsistente Weise ändern, ohne wissen zu müssen, wo genau die Änderung vorgenommen wird. Der Anfang des Bereichsobjekts dient dazu, die nächste gültige Position zu finden, an der die angegebene Aktion durchgeführt werden kann. Wenn es keine solche Position gibt, gibt die Methode den Wert *false* zurück. Andernfalls wird das Dokument geändert und der Bereich zeigt auf den Anfang der Änderung.

HINWEIS: Um Elementnamen zu finden, die als der zweite Parameter übergeben werden können, verwenden Sie CanPerformActionWith.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.
2007	Es wurde eine ungültige Aktion definiert.

Beispiele

```
' ------------------------------------------
'             VBScript
' Insert the innermost element
' ------------------------------------------
Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' we determine the elements that can be inserted at the current position
Dim arrElements()
objRange.CanPerformActionWith spyAuthenticInsertBefore, arrElements

' we insert the first (innermost) element
If UBound(arrElements) >= 0 Then
     objRange.PerformAction spyAuthenticInsertBefore, arrElements(0)
     ' objRange now points to the beginning of the inserted element
     ' we set a default value and position at its end
     objRange.Text = "Hello"
     objRange.ExpandTo(spyAuthenticTag).CollapsToEnd().Select
Else
     MsgBox "Can't insert any elements at current position"
End If
```


© 2010 Altova GmbH

2.7.42 AuthenticRange.Select

Siehe auch

Methode: Select ()

Beschreibung
Macht diesen Bereich zur aktuellen Auswahl der Ansicht. Dasselbe Ergebnis erzielen Sie bei Verwendung von: '*objRange.Parent.Selection* = *objRange*'

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.

Beispiele

```
' ----------------------------------------
'                   VBScript
' ----------------------------------------

' set current selection to end of document
objPlugin.objAuthenticView.DocumentEnd.Select()
```

2.7.43 AuthenticRange.SelectNext

Siehe auch

Methode: SelectNext (*eKind* als SPYAuthenticElementKind) als AuthenticRange

Beschreibung
Wählt das Element vom Typ *eKind* nach dem aktuellen Bereichsende aus. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'                  VBScript
' Scan through the whole document word-by-word
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

Dim objRange
Set objRange = objAuthenticView.DocumentBegin
Dim bEndOfDocument
bEndOfDocument = False

On Error Resume Next
While Not bEndOfDocument
     objRange.SelectNext(spyAuthenticWord).Select
     If ((Err.number - vbObjecterror) = 2003) Then
            bEndOfDocument = True
            Err.Clear
     ElseIf (Err.number <> 0) Then
            Err.Raise  ' forward error
     End If
Wend
```


© 2010 Altova GmbH

2.7.44 AuthenticRange.SelectPrevious

Siehe auch

Methode: GotoPrevious (*eKind* als SPYAuthenticElementKind) als AuthenticRange

Beschreibung
Wählt das Element vom Typ *eKind* vor dem Beginn es aktuelle Bereichs aus. Die Methode gibt das geänderte Bereichsobjekt zurück.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2004	Das Ziel befindet vor dem Beginn des Dokuments.
2005	Es wurde ein ungültiger Elementtyp definiert. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ------------------------------------------
'                VBScript
' Scan through the whole document tag-by-tag
' ------------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

Dim objRange
Set objRange = objAuthenticView.DocumentEnd
Dim bEndOfDocument
bBeginOfDocument = False

On Error Resume Next
While Not bBeginOfDocument
     objRange.SelectPrevious(spyAuthenticTag).Select
     If ((Err.number - vbObjecterror) = 2004) Then
            bBeginOfDocument = True
            Err.Clear
     ElseIf (Err.number <> 0) Then
            Err.Raise  ' forward error
     End If
Wend
```

2.7.45 AuthenticRange.SetElementAttributeValue

Siehe auch

Methode: SetElementAttributeValue (*strElementName* als String, *strAttributeName* als String, *strAttributeValue* als String)

Beschreibung
Liefert den Wert des in *strAttributeName* Attributs für das mit *strElementName* bezeichnete Element. Wenn das Attribut unterstützt wird, aber keinen zugewiesenen Wert hat, wird ein leerer String zurückgegeben. Um die Namen der von einem Element unterstützten Attribute zu ermitteln, verwenden Sie GetElementAttributeNames oder HasElementAttribute.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Elementname angegeben. Es wurde ein ungültiger Attributname angegeben. Es wurde ein ungültiger Attributwert angegeben.

Beispiele

```
' ------------------------------------------
'            VBScript
' Get and set element attributes
' ------------------------------------------
Dim objRange
Set objRange = objPlugin.AuthenticView.Selection

' first we find out all the elements below the beginning of the range
Dim arrElements
objRange.GetElementHierarchy arrElements

If IsArray(arrElements) Then
     If UBound(arrElements) >= 0 Then
            ' we use the top level element and find out its valid attributes
            Dim arrAttrs()
            objRange.GetElementAttributeNames arrElements(0), arrAttrs

            If UBound(arrAttrs) >= 0 Then
                  ' we retrieve the current value of the first valid
attribute
                  Dim strAttrVal
                  strAttrVal = objRange.GetElementAttributeValue
(arrElements(0), arrAttrs(0))
                  msgbox "current value of " & arrElements(0) & "//" &
arrAttrs(0) & " is: " & strAttrVal

                  ' we change this value and read it again
                  strAttrVal = "Hello"
                  objRange.SetElementAttributeValue arrElements(0),
arrAttrs(0), strAttrVal
                  strAttrVal = objRange.GetElementAttributeValue
(arrElements(0), arrAttrs(0))
                  msgbox "new value of " & arrElements(0) & "//" &
arrAttrs(0) & " is: " & strAttrVal
            End If
     End If
End If
```


© 2010 Altova GmbH

2.7.46 AuthenticRange.SetFromRange

Siehe auch

Methode: SetFromRange (*objSrcRange* als AuthenticRange)

Beschreibung
Setzt das Bereichsobjekt auf dieselbe Anfangs- und Endposition wie *objSrcRange*.

Fehler
2001 Eines der beiden Bereichsobjekte ist ungültig.
2005 Es wurde kein Objekt als Quellobjekt definiert.

2.7.47 AuthenticRange.Text

Siehe auch

Eigenschaft: Text als String

Beschreibung
Setzt oder holt den vom Bereichsobjekt ausgewählten Textinhalt.

Die Anzahl der abgerufenen Zeichen muss nicht unbedingt übereinstimmen, da es zwischen dem Anfang und dem Ende des ausgewählten Bereichs Textcursor-Positionen gibt. Die meisten Dokumentelemente unterstützen eine Bereichsendeposition des Cursors, die nicht mit der Bereichsanfangsposition des folgenden Elements übereinstimmt. Dropdown-Listen verfügen nur über eine Cursorposition, können aber Strings beliebiger Länge auswählen. Bei Optionsfeldern und Kontrollkästchen enthält der Texteigenschaftswert den String des entsprechenden XML-Elements.

Wenn mit dem Bereich mehr als ein Element auswählt wird, besteht der Text aus einer Verkettung der Einzeltexte. XML Entities werden erweitert, so dass statt '&' '&' erscheint.

Wenn der Text auf den leeren String gesetzt wird, wird kein XML-Element gelöscht. Verwenden Sie stattdessen Cut, Delete oder PerformAction.

Fehler

2001	Das Authentic Range-Objekt oder das dazugehörige View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.8 AuthenticSelection

Siehe auch

Eigenschaften
Start
StartTextPosition
End
EndTextPosition

2.8.1 AuthenticSelection.End

Siehe auch

Deklaration: End als XMLData

Beschreibung
XML-Element, an dem die aktuelle Auswahl endet.

© 2010 Altova GmbH

2.8.2 AuthenticSelection.EndTextPosition

Siehe auch

Deklaration: EndTextPosition als long

Beschreibung
Position in Authentic.End, an der die Auswahl endet.

2.8.3 AuthenticSelection.Start

Siehe auch

Deklaration: Start als XMLData

Beschreibung
XML-Element, an dem die aktuelle Auswahl beginnt.

© 2010 Altova GmbH

2.8.4 AuthenticSelection.StartTextPosition

Siehe auch

Deklaration: StartTextPosition als long

Beschreibung
Position in Authentic.Start, an der die Auswahl beginnt.

2.9 AuthenticToolbarButton

Methoden

Eigenschaften

CommandID

© 2010 Altova GmbH

2.9.1 AuthenticToolbarButton.CommandID

Deklaration: CommandID als SPYAuthenticCommand

Beschreibung
Die CommandId der Symbolleisten-Schaltfläche.

2.10 AuthenticToolbarButtons

Methoden

Item

NewButton
NewCustomButton
NewSeparator

Remove

Eigenschaften

Count

© 2010 Altova GmbH

2.10.1 AuthenticToolbarButtons.Count

Deklaration: Count als long

Beschreibung
Holt die tatsächliche Anzahl der Schaltflächen der Symbolleiste.
Schreibgeschützt.

2.10.2 AuthenticToolbarButtons.Item

Deklaration: Item(n als long) als AuthenticToolbarButton

Beschreibung
Holt die nte Schaltfläche der aktuellen Symbolleiste. n beginnt mit 1.

Beispiel
Siehe Beispiel unter AuthenticToolbarButtons.NewButton

© 2010 Altova GmbH

2.10.3 AuthenticToolbarButtons.NewButton

Deklaration: NewButton(nPosition als long, nCommandId als SPYAuthenticCommand)

Beschreibung
Fügt eine neue Schaltfläche für den Befehl "nCommandId" an der Position "nPosition" der Symbolleiste ein. nPosition beginnt mit 1.

Beispiel
Fügt eine neue Symbolleiste hinzu, ordnet sie am unteren Rand an und fügt neue Symbolleisten-Schaltflächen hinzu.

```
objPlugIn.ToolbarRows.NewRow(3)                    // add a new Toolbar (Row
                                                   3)
set ToolbarRow  = objPlugIn.ToolbarRows.Item(3)
set Buttons = ToolbarRow.Buttons
ToolbarRow.Alignment    = 2          // align Toolbar to bottom
Buttons.NewButton       1, 2         // add Print Button
Buttons.NewButton       1, 3         // add PrintPreview Button
Buttons.NewSeparator    2            // add Separator
Buttons.NewButton       1, 4         // add Validate Button
```

Bei Aufruf von StartEditing oder ReloadToolbars werden die geänderten Symbolleisteneinstellungen verwendet.

2.10.4 AuthenticToolbarButtons.NewCustomButton

Deklaration: NewCustomButton(nPosition als long, strName als String, strTooltip als String, strBitmapURL als String)

Beschreibung
Fügt in der Symbolleiste an der Position nPosition eine neue benutzerdefinierte Schaltfläche mit dem Namen strName ein. nPosition beginnt bei 1.
strTooltip wird als Text für den Tooltipp verwendet.

strBitmapURL ist der Pfad zur Bitmap-Datei für die neue Schaltfläche. Dieser Pfad ist relativ zum Pfad, der mit der TextStateBmpURL-Eigenschaft definiert wurde.

Beispiel

Angenommen, Sie haben eine benutzerdefinierte Schaltfläche mit dem Namen "MyFunction" in Ihre Symbolleisten eingefügt. Im folgenden Event Handler für doceditcommand sehen Sie, was geschehen muss, wenn der Benutzer auf Ihre Schaltfläche geklickt hat und wie Sie den ein/aus-Status dafür festlegen.

```
<SCRIPT LANGUAGE=javascript FOR=objPlugIn EVENT=doceditcommand>
 // event.type is set to "command" if the user clicked the button
 if( objPlugIn.event.type == "command")
 {
  if( objPlugIn.event.srcElement.Name == "MyFunction")
   window.alert("You pressed my custom button!");
 }

 // event.type is set to "update" if the button state must be set
 if( objPlugIn.event.type == "update")
 {
  if( objPlugIn.event.srcElement.Name == "MyFunction")
  {
   // we enable the button if only one element is selected
   if
( objPlugIn.CurrentSelection.Start.IsSameNode( objPlugIn.CurrentSelection.End))
    objPlugIn.event.returnValue = N;
   else
    objPlugIn.event.returnValue = 0;

   objPlugIn.event.cancelBubble = true;
  }
 }
</SCRIPT>
```


© 2010 Altova GmbH

2.10.5 AuthenticToolbarButtons.NewSeparator

Deklaration: NewSeparator(nPosition als long)

Beschreibung
Fügt ein Trennzeichen an der Position nPosition der Symbolleiste ein. nPosition beginnt mit 1.

Beispiel
Siehe Beispiel unter AuthenticToolbarButtons.NewButton

2.10.6 AuthenticToolbarButtons.Remove

Deklaration: Remove(nPosition als long)

Beschreibung
Entfernt die Schaltfläche oder das Trennzeichen an Position nPosition der Symbolleiste. nPosition beginnt bei 1.

© 2010 Altova GmbH

2.11 AuthenticToolbarRow

Methoden

Alignment

Eigenschaften

Buttons

2.11.1 AuthenticToolbarRowAlignment

Deklaration: Alignment(nAlign als SPYAuthenticToolbarAllignment)

Beschreibung
Liefert oder setzt die Anordnung der Symbolleiste im Plug-In.

Beispiel
Anordnen aller Symbolleisten am unteren Rand:

```
for each ToolbarRow in objPlugin.ToolbarRows
 ToolbarRow.Alignment = 2
next
```


© 2010 Altova GmbH

2.11.2 AuthenticToolbarRowButtons

Deklaration: Buttons als AuthenticToolbarButtons

Beschreibung
Liefert alle Schaltflächen der Symbolleiste

Beispiel
Siehe Beispiel unter AuthenticToolbarButtons.NewButton

2.12 AuthenticToolbarRows

Methoden

Item
RemoveRow
NewRow

Eigenschaften

Count

© 2010 Altova GmbH

2.12.1 AuthenticToolbarRows.Count

Deklaration: Count als long

Beschreibung
Liefert die aktuelle Anzahl der definierten Symbolleisten.
Schreibgeschützt.

2.12.2 AuthenticToolbarRows.Item

Deklaration: Item(nPosition als long) als AuthenticToolbarRow

Beschreibung
Liefert die Symbolleiste in Position nPosition. nPosition beginnt mit "1".
Schreibgeschützt.

Beispiel
Siehe Beispiel unter AuthenticToolbarButtons.NewButton

© 2010 Altova GmbH

2.12.3 AuthenticToolbarRows.RemoveRow

Deklaration: RemoveRow(nPosition als long)

Beschreibung
Entfernt die Symbolleiste an Position nPosition von der Benutzeroberfläche. nPosition beginnt mit 1.

2.12.4 AuthenticToolbarRows.NewRow

Deklaration: NewRow(nPosition als long)

Beschreibung
Fügt eine neue Symbolleistenzeile ein. nPosition beginnt bei 1.

Beispiel
Siehe Beispiel unter AuthenticToolbarButtons.NewButton

© 2010 Altova GmbH

2.13 AuthenticView

Siehe auch

Eigenschaften	Methoden	Ereignisse
Application	Goto	
DocumentBegin	Print	
DocumentEnd	Redo	
MarkupVisibility	Undo	
Parent	UpdateXMLInstanceEntities	
Selection		
WholeDocument		

Beschreibung
AuthenticView und sein Child-Objekt AuthenticRange stellen die Schnittstelle für die Authentic -Ansicht bereit, über die Sie Dokument-Inhalte einfach und auf konsistente Art bearbeiten können. Die Funktionsüberschneidung mit bereits bestehenden Methoden und Eigenschaften im Authentic-Objekt ist beabsichtigt. Die Funktionen des Authentic-Objekts zur Bearbeitung von Inhalt werden dadurch nicht mehr benötigt. Es wird empfohlen, unbedingt die neue AuthenticView-Schnittstelle zu verwenden.

AuthenticView verschafft Ihnen einfachen Zugriff auf bestimmte Funktionen wie z.B. Drucken, mehrfaches Rückgängigmachen und die aktuelle Cursor-Auswahl oder -Position.

AuthenticView verwendet Objekte vom Typ AuthenticRange, um das einfache Navigieren im Dokument und die problemlose Auswahl logischer Textelemente zu ermöglichen. Verwenden Sie für einfache Auswahlvorgänge die Eigenschaften DocumentBegin, DocumentEnd oder WholeDocument und für komplexere Auswahlvorgänge die Goto-Methode. Informationen zum Navigieren relativ zu einem vorgegebenen Dokumentenbereich finden Sie unter den Methoden und Eigenschaften des AuthenticRange-Objekts.

2.13.1 AuthenticView.Application

Siehe auch

Eigenschaft: Application als Authentic (Schreibgeschützt)

Beschreibung
Ruft das Authentic Desktop Applikationsobjekt auf.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.13.2 AuthenticView.DocumentBegin

Siehe auch

Eigenschaft: DocumentBegin als AuthenticRange (Schreibgeschützt)

Beschreibung
Ruft ein Bereichsobjekt ab, das auf den Beginn des Dokuments zeigt.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.13.3 AuthenticView.DocumentEnd

Siehe auch

Eigenschaft: DocumentEnd als AuthenticRange (Schreibgeschützt)

Beschreibung
Ruft ein Bereichsobjekt ab, das auf das Ende des Dokuments zeigt.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.13.4 AuthenticView.Goto

Siehe auch

Methode: Goto (*eKind* als SPYAuthenticElementKind, *nCount* als Long, *eFrom* als SPYAuthenticDocumentPosition) als AuthenticRange

Beschreibung
Ruft ein Bereichsobjekt auf, das auf den Anfang des *nCount*-Elements vom Typ *eKind* zeigt. Die Anfangsposition wird durch den Parameter *eFrom* definiert. Verwenden Sie positive Werte für *nCount*, um zum Ende des Dokuments zu navigieren. Verwenden Sie negative Werte, um zum Anfang des Dokuments zu navigieren.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2003	Das Ziel befindet sich hinter dem Ende des Dokuments.
2004	Das Ziel befindet vor den Anfang des Dokuments.
2005	Es wurde ein ungültiger Elementtyp angegeben. Die Anfangsposition ist keine Position aus *spyAuthenticDocumentBegin* oder *spyAuthenticDocumentEnd*. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

On Error Resume Next
Dim objRange
' goto beginning of first table in document
Set objRange = objAuthenticView.Goto (spyAuthenticTable, N,
spyAuthenticDocumentBegin)
If (Err.number = 0) Then
     objRange.Select()
Else
     MsgBox "No table found in document"
End If
```

2.13.5 AuthenticView.MarkupVisibility

Siehe auch

Eigenschaft: MarkupVisibility als SPYAuthenticMarkupVisibility

Beschreibung
Setzt oder liefert den aktuellen Sichtbarkeitsstatus von Markup.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Es wurde ein ungültiger Enumerationswert angegeben. Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

© 2010 Altova GmbH

2.13.6 AuthenticView.Parent

Siehe auch

Eigenschaft: Parent als Authentic (schreibgeschützt)

Beschreibung
Liefert eine Referenz auf die Applikation.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.13.7 AuthenticView.Print

Siehe auch

Methode: Print (*bWithPreview* als Boolean, *bPromptUser* als Boolean)

Beschreibung
Druckt das in dieser Ansicht angezeigte Dokument. Wenn *bWithPreview* auf *True* gesetzt ist, wird das Dialogfeld "Druckvorschau" angezeigt. Wenn *bPromptUser* auf *True* gesetzt ist, wird das Dialogfeld "Drucken" angezeigt. Wenn beide Parameter auf *False* gesetzt sind, wird das Dokument ohne weiteren Eingriff durch den Benutzer gedruckt.

Fehler
2000 Das Authentic View-Objekt ist nicht mehr gültig.

© 2010 Altova GmbH

2.13.8 AuthenticView.Redo

Siehe auch

Methode: Redo () als Boolean

Beschreibung
Stellt die letzte mit dem Befehl "Rückgängig" rückgängig gemachte Aktion wieder her.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.13.9 AuthenticView.Selection

Siehe auch

Eigenschaft: Selection als AuthenticRange

Beschreibung
Setzt oder holt die aktuelle Textauswahl auf der Benutzeroberfläche.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2002	Es ist keine Selektion aktiv.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

Beispiele

```
' ----------------------------------------
'               VBScript
' ----------------------------------------
Dim objAuthenticView
Set objAuthenticView = objPlugin.AuthenticView

' if we are the end of the document, re-start at the beginning
If (objAuthenticView.Selection.IsEqual(objAuthenticView.DocumentEnd)) Then
     objAuthenticView.Selection = objAuthenticView.DocumentBegin
Else
     ' objAuthenticView.Selection =
objAuthenticView.Selection.GotoNextCursorPosition()
     ' or shorter:
      objAuthenticView.Selection.GotoNextCursorPosition().Select
End If
```


© 2010 Altova GmbH

2.13.10 AuthenticView.Undo

Siehe auch

Methode: Undo () als Boolean

Beschreibung
Macht die letzte in dieser Ansicht vorgenommene Änderung am Dokument rückgängig.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.13.11 AuthenticView.UpdateXMLInstanceEntities

Siehe auch

Methode: UpdateXMLInstanceEntities ()

Beschreibung
Aktualisiert die interne Darstellung der deklarierten Entities und befüllt die Eingabehilfen erneut. Zusätzlich dazu wird der Validator neu geladen, damit die XML-Datei korrekt validiert werden kann. Beachten Sie, dass dadurch eventuell auch Schema-Dateien neu geladen werden.

Fehler
Die Methode gibt nie einen Fehler zurück.

Beispiel

```
// ----------------------------------------
//               JavaScript
// ----------------------------------------
var objDocType;
objDocType = objPlugin.XMLRoot.GetFirstChild(N0);

if(objDocType)
{
     var objEntity = objPlugin.CreateChild(N4);
     objEntity.Name = "child";
     objEntity.TextValue = "SYSTEM \"child.xml\"";
     objDocType.AppendChild(objEntity);

     objPlugin.AuthenticView.UpdateXMLInstanceEntities();
}
```


© 2010 Altova GmbH

2.13.12 AuthenticView.WholeDocument

Siehe auch

Eigenschaft: WholeDocument als AuthenticRange (Schreibgeschützt)

Beschreibung
Ruft ein Bereichsobjekt ab, das das gesamte Dokument auswählt.

Fehler

2000	Das Authentic View-Objekt ist nicht mehr gültig.
2005	Für den zurückgegebenen Parameter wurde eine ungültige Adresse angegeben.

2.14 AuthenticXMLTableCommands

Methoden

Insert
Delete
AppendRow
InsertRow
DeleteRow
AppendColumn
InsertColumn
DeleteColumn
JoinLeft
JoinRight
JoinUp
JoinDown
SplitHorizontal
SplitVertical
AlignVerticalTop
AlignVerticalCenter
AlignVerticalBottom
AlignHorizontalLeft
AlignHorizontalRight
AlignHorizontalCenter
AlignHorizontalJustify
EditProperties

Eigenschaften

MayInsert
MayDelete
MayAppendRow
MayInsertRow
MayDeleteRow
MayAppendCol
MayInsertCol
MayDeleteCol
MayJoinLeft
MayJoinRight
MayJoinUp
MayJoinDown
MaySplitHorizontal
MaySplitVertical

© 2010 Altova GmbH

MayAlignVertical
MayAlignHorizontal
MayEditProperties

2.14.1 AuthenticXMLTableCommands.AlignHorizontalCenter

Deklaration: AlignHorizontalCenter()

Beschreibung
Setzt die horizontale Ausrichtung auf "zentriert" oder setzt das Attribut zurück, wenn es zuvor auf "zentriert" gesetzt war.

© 2010 Altova GmbH

2.14.2 AuthenticXMLTableCommands.AlignHorizontalJustify

Deklaration: AlignHorizontalJustify()

Beschreibung
Setzt die horizontale Ausrichtung auf "Blocksatz" oder setzt das Attribut zurück, wenn es zuvor auf "Blocksatz" gesetzt war.

2.14.3 AuthenticXMLTableCommands.AlignHorizontalLeft

Deklaration: AlignHorizontalLeft()

Beschreibung
Setzt die horizontale Ausrichtung auf "links" oder setzt das Attribut zurück, wenn es zuvor auf "links" gesetzt war.

© 2010 Altova GmbH

2.14.4 AuthenticXMLTableCommands.AlignHorizontalRight

Deklaration: AlignHorizontalRight()

Beschreibung
Setzt die horizontale Ausrichtung auf "rechts" oder setzt das Attribut zurück, wenn es zuvor auf "rechts" gesetzt war.

2.14.5 AuthenticXMLTableCommands.AlignVerticalBottom

Deklaration: AlignVerticalBottom()

Beschreibung
Setzt die vertikale Ausrichtung auf "unten" oder setzt das Attribut zurück, wenn es zuvor auf "unten" gesetzt war.

© 2010 Altova GmbH

2.14.6 AuthenticXMLTableCommands.AlignVerticalCenter

Deklaration: AlignVerticalCenter()

Beschreibung
Setzt die vertikale Ausrichtung auf "Mitte" oder setzt das Attribut zurück, wenn es zuvor auf "Mitte" gesetzt war.

2.14.7 AuthenticXMLTableCommands.AlignVerticalTop

Deklaration: AlignVerticalTop()

Beschreibung
Setzt die vertikale Ausrichtung auf "oben" oder setzt das Attribut zurück, wenn es zuvor auf "oben" gesetzt war.

© 2010 Altova GmbH

2.14.8 AuthenticXMLTableCommands.AppendCol

Deklaration: AppendCol()

Beschreibung
Hängt eine Spalte an die aktuelle Tabelle an.

2.14.9 AuthenticXMLTableCommands.AppendRow

Deklaration: AppendRow()

Beschreibung
Hängt eine Zeile an die aktuelle Tabelle an.

© 2010 Altova GmbH

2.14.10 AuthenticXMLTableCommands.Delete

Deklaration: Delete()

Beschreibung
Wenn der Benutzer den Vorgang bestätig hat, löscht die Methode die aktuell ausgewählte Tabelle.

2.14.11 AuthenticXMLTableCommands.DeleteCol

Deklaration: DeleteCol()

Beschreibung
Die Methode löscht die aktuell ausgewählte Spalte.
Wenn es nur mehr eine Spalte gibt und der Benutzer den Vorgang bestätigt, wird die gesamte Tabelle gelöscht.

© 2010 Altova GmbH

2.14.12 AuthenticXMLTableCommands.DeleteRow

Deklaration: DeleteRow()

Beschreibung
Die Methode löscht die aktuell ausgewählte Zeile.
Wenn es nur mehr eine Zeile gibt und der Benutzer den Vorgang bestätigt, wird die gesamte Tabelle gelöscht.

2.14.13 AuthenticXMLTableCommands.EditProperties

Deklaration: EditProperties()

Beschreibung
Zeigt das Dialogfeld "Eigenschaften" für die ausgewählte Tabelle/Zelle an.

© 2010 Altova GmbH

2.14.14 AuthenticXMLTableCommands.Insert

Deklaration: Insert()

Beschreibung
Ruft ein Dialogfeld auf und fügt eine neue Tabelle in die bestehende XML-Datei ein.

2.14.15 AuthenticXMLTableCommands.InsertCol

Deklaration: InsertCol()

Beschreibung
Fügt eine neue Spalte vor der aktuelle ausgewählten Spalte ein.

© 2010 Altova GmbH

2.14.16 AuthenticXMLTableCommands.InsertRow

Deklaration: InsertRow()

Beschreibung
Fügt eine Zeile vor der aktuell ausgewählten Zeile ein.

2.14.17 AuthenticXMLTableCommands.JoinDown

Deklaration: JoinDown()

Beschreibung
Verbindet die aktuelle Zelle mit der Zelle unmittelbar darunter.

© 2010 Altova GmbH

2.14.18 AuthenticXMLTableCommands.JoinLeft

Deklaration: JoinLeft()

Beschreibung
Verbindet die aktuelle Zelle mit der Zelle unmittelbar links davon.

2.14.19 AuthenticXMLTableCommands.JoinRight

Deklaration: JoinRight()

Beschreibung
Verbindet die aktuelle Zelle mit der Zelle unmittelbar rechts davon.

© 2010 Altova GmbH

2.14.20 AuthenticXMLTableCommands.JoinUp

Deklaration: JoinUp()

Beschreibung
Verbindet die aktuelle Zelle mit der Zelle unmittelbar darüber.

2.14.21 AuthenticXMLTableCommands.MayAlignHorizontal

Deklaration: MayAlignHorizontal als Boolean

Beschreibung
True, wenn die Attribute für die horizontale Ausrichtung für die aktuelle Zelle gesetzt werden können.

© 2010 Altova GmbH

2.14.22 AuthenticXMLTableCommands.MayAlignVertical

Deklaration: MayAlignVertical als Boolean

Beschreibung
True, wenn die Attribute für die vertikale Ausrichtung für die aktuelle Zelle gesetzt werden können.

2.14.23 AuthenticXMLTableCommands.MayAppendCol

Deklaration: MayAppendCol als Boolean

Beschreibung
True, wenn eine Spalte angehängt werden kann.

© 2010 Altova GmbH

2.14.24 AuthenticXMLTableCommands.MayAppendRow

Deklaration: MayAppendRow als Boolean

Beschreibung
True, wenn eine Zeile angehängt werden kann.

2.14.25 AuthenticXMLTableCommands.MayDelete

Deklaration: MayDelete als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn eine Tabelle ausgewählt ist.

© 2010 Altova GmbH

2.14.26 AuthenticXMLTableCommands.MayDeleteCol

Deklaration: MayDeleteCol als Boolean

Beschreibung
True, wenn die aktuelle Spalte gelöscht werden kann.

2.14.27 AuthenticXMLTableCommands.MayDeleteRow

Deklaration: MayDeleteRow als Boolean

Beschreibung
True, wenn die aktuelle Zeile gelöscht werden kann.

© 2010 Altova GmbH

2.14.28 AuthenticXMLTableCommands.MayEditProperties

Deklaration: MayEditProperties als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn das Dialogfeld "Eigenschaften" für die ausgewählte Tabelle/Zelle zur Verfügung steht.

2.14.29 AuthenticXMLTableCommands.MayInsert

Deklaration: MayInsert als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn an der aktuellen Auswahl eine neue Tabelle eingefügt werden kann.

© 2010 Altova GmbH

2.14.30 AuthenticXMLTableCommands.MayInsertCol

Deklaration: MayInsertCol als Boolean

Beschreibung
True, wenn eine Spalte eingefügt werden kann.

2.14.31 AuthenticXMLTableCommands.MayInsertRow

Deklaration: MayInsertRow als Boolean

Beschreibung
True, wenn eine Zeile eingefügt werden kann.

© 2010 Altova GmbH

2.14.32 AuthenticXMLTableCommands.MayJoinDown

Deklaration: MayJoinDown als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn die JoinDown-Operation für die aktuell ausgewählte Zelle möglich ist.

2.14.33 AuthenticXMLTableCommands.MayJoinLeft

Deklaration: MayJoinLeft als Boolean

Beschreibung
Die Eigenschaft ist true, wenn die JoinLeft-Operation für die aktuell ausgewählte Zelle möglich ist.

© 2010 Altova GmbH

2.14.34 AuthenticXMLTableCommands.MayJoinRight

Deklaration: MayJoinRight als Boolean

Beschreibung
Die Eigenschaft ist true, wenn die JoinRight-Operation für die aktuell ausgewählte Zelle möglich ist.

2.14.35 AuthenticXMLTableCommands.MayJoinUp

Deklaration: MayJoinUp als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn die JoinUp-Operation für die aktuell ausgewählte Zelle möglich ist.

© 2010 Altova GmbH

2.14.36 AuthenticXMLTableCommands.MaySplitHorizontal

Deklaration: MaySplitHorizontal als Boolean

Beschreibung
True, wenn die aktuelle Zelle horizontal geteilt werden kann.

2.14.37 AuthenticXMLTableCommands.MaySplitVertical

Deklaration: MaySplitVertical als Boolean

Beschreibung
True, wenn die aktuelle Zelle vertikal geteilt werden kann.

© 2010 Altova GmbH

2.14.38 AuthenticXMLTableCommands.SplitHorizontal

Deklaration: SplitHorizontal()

Beschreibung
Die Methode teilt die aktuelle Zelle horizontal.

2.14.39 AuthenticXMLTableCommands.SplitVertical

Deklaration: SplitVertical()

Beschreibung
Die Methode teilt die aktuelle Zelle vertikal.

© 2010 Altova GmbH

2.15 XMLData

Siehe auch

Methoden
InsertChild
AppendChild

EraseAllChildren
EraseCurrentChild

GetCurrentChild
GetFirstChild
GetNextChild

GetChild
GetChildKind

IsSameNode

HasChildrenKind

CountChildren
CountChildrenKind

Eigenschaften
Name
TextValue

HasChildren
MayHaveChildren

Kind
Parent

Beschreibung
Mit Hilfe der XMLData-Schnittstelle können Sie den Inhalt der gerade angezeigten XML-Datei manipulieren. Die Schnittstelle ist ein vereinfachtes Pendant zur COM-Schnittstelle, die im Plug-In und in XMLSpy selbst implementiert ist.

Verwenden Sie zum Erstellen eines neuen XMLData-Objekts die CreateChild()-Methode der Plug-In-Schnittstelle.

2.15.1 XMLData.AppendChild

Siehe auch

Deklaration: AppendChild(*pNewData* als XMLData)

Beschreibung
AppendChild hängt pNewData als letztes Child an das XMLData-Objekt an. Siehe auch "Using XMLData".

Beispiel

```
Dim objCurrentParent
Dim objNewChild

Set objNewChild = objPlugIn.CreateChild(spyXMLDataElement)
Set objCurrentParent = objPlugIn.XMLRoot

objCurrentParent.AppendChild objNewChild

Set objNewChild = Nothing
```


© 2010 Altova GmbH

2.15.2 XMLData.CountChildren

Siehe auch

Deklaration: CountChildren als long

Beschreibung
`CountChildren` holt die Anzahl der Children.

Verfügbar mit TypeLibrary Version 1.6

Fehler
1500 Das XMLData-Objekt ist nicht mehr gültig.

2.15.3 XMLData.CountChildrenKind

Siehe auch

Deklaration: CountChildrenKind (*nKind* as SPYXMLDataKind) als long

Beschreibung

`CountChildrenKind` holt die Anzahl der Children der jeweiligen Art.

Verfügbar mit TypeLibrary Version 1.6

Fehler

1500 Das XMLData-Objekt ist nicht mehr gültig.

© 2010 Altova GmbH

2.15.4 XMLData.EraseAllChildren

Siehe auch

Deklaration: EraseAllChildren

Beschreibung
EraseAllChildren löscht alle verknüpften Children des XMLData-Objekts.

Beispiel

In diesem Beispiel werden alle Elemente des aktiven Dokuments gelöscht.

```
Dim objCurrentParent

Set objCurrentParent = objPlugIn.XMLRoot
objCurrentParent.EraseAllChildren
```

2.15.5 XMLData.EraseCurrentChild

Siehe auch

Deklaration: EraseCurrentChild

Beschreibung
EraseCurrentChild löscht das aktuelle XMLData Child-Objekt. Bevor Sie EraseCurrentChild aufrufen, müssen Sie mit XMLData.GetFirstChild einen internen Iterator initialisieren.

Beispiel

In diesem JavaScript-Beispiel werden alle Elemente mit dem Namen "EraseMe" gelöscht. Sie sehen im Code, dass innerhalb eines Schleifendurchlaufs EraseCurrentChild und GetNextChild aufgerufen werden können, sodass die Child-Elemente der Reihe nach abgearbeitet werden können.

```
function DeleteXMLElements(objXMLData)
{
 if(objXMLData == null)
  return;

 if(objXMLData.HasChildren){
  var objChild;
  objChild = objXMLData.GetFirstChild(-N);

  while(objChild) {
   DeleteXMLElements(objChild);

   try{
    if(objChild.Name == "EraseMe")
     objXMLData.EraseCurrentChild();

    objChild = objXMLData.GetNextChild();
   }
   catch(Err){
    objChild = null;
   }
  }
 }
}
```


© 2010 Altova GmbH

2.15.6 XMLData.GetChild

Siehe auch

Deklaration: `GetChild` (*`position`* as long) als XMLData

Rückgabewert
Gibt ein XML-Element als `XMLData`-Objekt zurück.

Beschreibung
`GetChild()` gibt eine Referenz auf das Child des jeweiligen Index (nullbasiert) zurück.

Verfügbar mit TypeLibrary Version 1.6

Fehler

1500	Das XMLData Objekt ist nicht mehr gültig.
1510	Es wurde eine ungültige Adresse für den Rückgabeparameter angegeben.

2.15.7 XMLData.GetChildKind

Siehe auch

Deklaration: GetChildKind (*position* as long, *nKind* as SPYXMLDataKind) als XMLData

Rückgabewert
Gibt ein XML-Element als `XMLData`-Objekt zurück.

Beschreibung
`GetChildKind()` gibt eine Referenz auf das Child dieser Art des jeweiligen Index (nullbasiert) zurück. Der Positionsparameter ist relativ zur Anzahl der Children der ngegebenen Art und nicht zu alle Children des Objekts.

Verfügbar mit TypeLibrary Version 1.6

Fehler

1500	Das XMLData Objekt ist nicht mehr gültig.
1510	Es wurde eine ungültige Adresse für den Rückgabeparameter angegeben.

© 2010 Altova GmbH

2.15.8 XMLData.GetCurrentChild

Siehe auch

Deklaration: GetCurrentChild als XMLData

Rückgabewert
Gibt ein XML-Element als XMLData-Objekt zurück.

Beschreibung
GetCurrentChild holt das aktuelle Child. Bevor Sie GetCurrentChild aufrufen, müssen Sie mit XMLData.GetFirstChild einen internen Iterator initialisieren.

2.15.9 XMLData.GetFirstChild

Siehe auch

Deklaration: GetFirstChild(*nKind* als SPYXMLDataKind) als XMLData

Rückgabewert
Gibt ein XML-Element als XMLData-Objekt zurück.

Beschreibung
GetFirstChild initialisiert einen neuen Iterator und gibt das erste Child zurück. Setzen Sie nKind = -1, um einen Iterator für alle Arten von Children zu holen.

Beispiel

Siehe Beispiel unter XMLData.GetNextChild.

© 2010 Altova GmbH

2.15.10 XMLData.GetNextChild

Siehe auch

Deklaration: GetNextChild als XMLData

Rückgabewert
Gibt ein XML-Element als XMLData-Objekt zurück.

Beschreibung
GetNextChild geht zum nächsten Child dieses Elements. Bevor Sie GetNextChild aufrufen, müssen Sie mit XMLData.GetFirstChild einen internen Iterator initialisieren.

Überprüft, wann das letzte Child des Elements erreicht wurde (siehe Beispiel unten).

Beispiel

```
On Error Resume Next
Set objParent = objPlugIn.XMLRoot

'get elements of all kinds
Set objCurrentChild = objParent.GetFirstChild(-N)

Do
 'do something useful with the child

 'step to next child
 Set objCurrentChild = objParent.GetNextChild
Loop Until (Err.Number - vbObjectError = N503)
```

2.15.11 XMLData.HasChildren

Siehe auch

Deklaration: HasChildren als Boolean

Beschreibung
Die Eigenschaft ist "true", wenn das Objekt ein Parent eines anderen XMLData-Objekts ist.

Diese Eigenschaft ist schreibgeschützt.

© 2010 Altova GmbH

2.15.12 XMLData.HasChildrenKind

Siehe auch

Dekaration: HasChildrenKind (*nKind* as SPYXMLDataKind) al Boolean

Description
Die Methode gibt "true" zurück, wenn das Objekt der Parent anderer `XMLData` Objekte der angegebenen Art ist.

Verfügbar mit TypeLibrary Version 1.6

Fehler

1500	Das XMLData Objekt ist nicht mehr gültig.
1510	Es wurde eine ungültige Adresse für den Rückgabeparameter angegeben.

2.15.13 XMLData.InsertChild

Siehe auch

Deklaration: InsertChild(*pNewData* als XMLData)

Beschreibung
InsertChild fügt das neue Child vor dem aktuellen Child ein (Siehe auch XMLData.GetFirstChild, XMLData.GetNextChild zu Setzen des aktuellen Child).

© 2010 Altova GmbH

2.15.14 XMLData.IsSameNode

Siehe auch

Deklaration: IsSameNode(*pNodeToCompare* als XMLData) als Boolean

Beschreibung
Gibt "true" zurück, wenn pNodeToCompare denselben Node referenziert wie das Objekt selbst.

2.15.15 XMLData.Kind

Siehe auch

Deklaration: Kind als SPYXMLDataKind

Beschreibung
Typ dieses XMLData-Objekts.

Diese Eigenschaft ist schreibgeschützt.

© 2010 Altova GmbH

2.15.16 XMLData.MayHaveChildren

Siehe auch

Deklaration: MayHaveChildren als Boolean

Beschreibung
Gibt an, ob es zulässig ist, Children zu diesem XMLData-Objekt hinzuzufügen.

Diese Eigenschaft ist schreibgeschützt.

2.15.17 XMLData.Name

Siehe auch

Deklaration: Name als String

Beschreibung
Wird verwendet, um den Namen des XMLData-Objekts zu holen und zu ändern.

© 2010 Altova GmbH

2.15.18 XMLData.Parent

Siehe auch

Deklaration: Parent als XMLData

Rückgabewert
Parent als XMLData-Objekt.
Kein Wert (oder NULL), wenn es kein Parent-Element gibt.

Beschreibung
Parent von diesem Element.

Diese Eigenschaft ist schreibgeschützt.

2.15.19 XMLData.TextValue

Siehe auch

Deklaration: TextValue als String

Beschreibung
Wird verwendet, um den Textwert von diesem XMLData-Objekt zu ändern und zu holen. Siehe auch "Using XMLData".

© 2010 Altova GmbH

3 Enumerationen

Dieser Abschnitt enthält eine Liste und Beschreibung der Authentic Browser Enumerationen.

3.1 SPYXMLDataKind

Beschreibung
Die verschiedenen Typen von XMLData-Elementen, die für XML-Dokumente zur Verfügung stehen

Mögliche Werte:

spyXMLDataXMLDocStruct	= 0
spyXMLDataXMLEntityDocStruct	= 1
spyXMLDataDTDDocStruct	= 2
spyXMLDataXML	= 3
spyXMLDataElement	= 4
spyXMLDataAttr	= 5
spyXMLDataText	= 6
spyXMLDataCData	= 7
spyXMLDataComment	= 8
spyXMLDataP	= 9
spyXMLDataDefDoctype	= 10
spyXMLDataDefExternalID	= 11
spyXMLDataDefElement	= 12
spyXMLDataDefAttlist	= 13
spyXMLDataDefEntity	= 14
spyXMLDataDefNotation	= 15
spyXMLDataKindsCount	= 16

© 2010 Altova GmbH

3.2 SPYAuthenticElementActions

Beschreibung
Aktionen, die mit GetAllowedElements verwendet werden können.

Mögliche Werte:

k_ActionInsertAt	= 0
k_ActionApply	= 1
k_ActionClearSurr	= 2
k_ActionAppend	= 3
k_ActionInsertBefore	= 4
k_ActionRemove	= 5

3.3 SPYAuthenticCommand

Beschreibung
Enumeration aller verfügbaren Befehle.

Mögliche Werte:

```
// CommandGroupMain
        k_CommandSeparator    = 0
        k_CommandSave         = 1
        k_CommandPrint        = 2
        k_CommandPrintPreview = 3
        k_CommandValidate     = 4
        k_CommandUndo         = 5
        k_CommandRedo         = 6

// CommandGroupEdit
        k_CommandEditCut      = 7
        k_CommandEditCopy     = 8
        k_CommandEditPaste    = 9
        k_CommandEditFind     = 10
        k_CommandEditRepeat   = 11
        k_CommandEditReplace  = 12

// CommandGroupMarkup
        k_CommandMarkupHide   = 13
        k_CommandMarkupLarge  = 14

// CommandGroupRow
        k_CommandRowAppend    = 15
        k_CommandRowInsert    = 16
        k_CommandRowDuplicate = 17
        k_CommandRowMoveUp    = 18
        k_CommandRowMoveDow   = 19
        n
        k_CommandRowDelete    = 20

// CommandGroupXMLTables
        k_CommandXMLTableInsert          = 21
        k_CommandXMLTableDelete          = 22
        k_CommandXMLTableAppendRow       = 23
        k_CommandXMLTableInsertRow       = 24
        k_CommandXMLTableDeleteRow       = 25
        k_CommandXMLTableAppendCol       = 26
        k_CommandXMLTableInsertCol       = 27
        k_CommandXMLTableDeleteCol       = 28
        k_CommandXMLTableJoinRight       = 29
        k_CommandXMLTableJoinLeft        = 30
        k_CommandXMLTableJoinUp          = 31
        k_CommandXMLTableJoinDown        = 32
        k_CommandXMLTableSplitHorz       = 33
        k_CommandXMLTableSplitVert       = 34
        k_CommandXMLTableVAlignTop       = 35
        k_CommandXMLTableVAlignCenter    = 36
        k_CommandXMLTableVAlignBottom    = 37
        k_CommandXMLTableAlignLeft       = 38
        k_CommandXMLTableAlignCenter     = 39
```


© 2010 Altova GmbH

```
            k_CommandXMLTableAlignRight      = 40
            k_CommandXMLTableAlignJustify    = 41
            k_CommandXMLTableEditProperties= 42

    // seit TypeLib 1.2
            k_CommandCheckSpelling           = 43
            k_CommandAbout                   = 44
            k_CommandPackageManagement       = 45
```

3.4 SPYAuthenticCommandGroup

Beschreibung
Gruppen, zu denen ein Befehl gehören kann.

Mögliche Werte:

k_CommandGroupMain	= 0
k_CommandGroupEdit	= 1
k_CommandGroupMarkup	= 2
k_CommandGroupRow	= 3
k_CommandGroupXMLTables	= 4

© 2010 Altova GmbH

3.5 SPYAuthenticToolbarAllignment

Beschreibung
Werte zur Definition der Symbolleistenposition.

Mögliche Werte:

k_ToolbarAlignTop	= 0
k_ToolbarAlignLeft	= 1
k_ToolbarAlignBottom	= 2
k_ToolbarAlignRight	= 3

3.6 SPYAuthenticEntryHelperWindows

Beschreibung
Identifizierung der verfügbaren Eingabehilfenfenster

Mögliche Werte:

k_Elements	= 1
k_Attributes	= 2
k_Entities	= 4

© 2010 Altova GmbH

3.7 SPYAuthenticElementKind

Beschreibung
Enumeration der verschiedenen zur Navigation und Auswahl innerhalb der Objekte AuthenticRange und AuthenticView verwendeten Elementarten.

Mögliche Werte:

spyAuthenticChar	= 0
spyAuthenticWord	= 1
spyAuthenticLine	= 3
spyAuthenticParagraph	= 4
spyAuthenticTag	= 6
spyAuthenticDocument	= 8
spyAuthenticTable	= 9
spyAuthenticTableRow	= 10
spyAuthenticTableColumn	= 11

3.8 SPYAuthenticActions

Beschreibung
Aktionen, die an AuthenticRange-Objekten ausgeführt werden können.

Mögliche Werte:

spyAuthenticInsertAt	= 0
spyAuthenticApply	= 1
spyAuthenticClearSurr	= 2
spyAuthenticAppend	= 3
spyAuthenticInsertBefore	= 4
spyAuthenticRemove	= 5

© 2010 Altova GmbH

3.9 SPYAuthenticDocumentPosition

Beschreibung
Relative und absolute Position zum Navigieren mit AuthenticRange-Objekten.

Mögliche Werte:

spyAuthenticDocumentBegin = 0
spyAuthenticDocumentEnd = 1
spyAuthenticRangeBegin = 2
spyAuthenticRangeEnd = 3

3.10 SPYAuthenticMarkupVisibility

Beschreibung
Enumerationswerte zum Aus- oder Einblenden des Markups.

Mögliche Werte:

spyAuthenticMarkupHidden = 0
spyAuthenticMarkupSmall = 1
spyAuthenticMarkupLarge = 2
spyAuthenticMarkupMixed = 3

© 2010 Altova GmbH

Kapitel 8

Lizenzinformationen

Lizenzinformationen

Dieser Abschnitt enthält die folgenden Informationen:

- Informationen über den Vertrieb dieses Software-Produkts
- Copyright-Informationen im Zusammenhang mit diesem Software-Produkt
- Die Endbenutzer-Lizenzvereinbarung (EULA) zu diesem Software-Produkt

Lesen Sie die Informationen bitte sorgfältig - sie sind rechtlich bindend, da Sie sich bei der Installation dieses Software-Produkts damit einverstanden erklärt haben.

© 2010 Altova GmbH

1 Electronic Software Distribution

Dieses Produkt ist über EDS (Electronic Software Distribution), also auf elektronischem Weg erhältlich, eine Methode, die die folgenden einzigartigen Vorteile bietet:

- Sie können die Software kostenlos testen, bevor Sie sich zu einem Kauf entscheiden.
- Wenn Sie sich entschieden haben, die Software zu kaufen, können Sie Ihre Bestellung online auf der Altova Website abgeben und das Produkt steht Ihnen in Minutenschnelle im vollen Lizenzumfang zur Verfügung.
- Sie erhalten immer die neueste Version unserer Software
- Die Software enthält ein umfassendes Online-Hilfesystem. Die neueste Version des Benutzerhandbuchs steht auf unserer Website unter http://www.altova.com/support_help.html (i) im HTML-Format zum Aufrufen online und (ii) im PDF-Format zum Download und Ausdrucken zur Verfügung.

30-Tage-Evaluierungszeitraum
Nachdem Sie dieses Software-Produkt heruntergeladen haben, können Sie es 30 Tage lang kostenlos testen. Während dieses Zeitraums werden Sie nach etwa 20 Tagen in regelmäßigen Abständen daran erinnert, dass die Software noch nicht lizenziert wurde. Diese Erinnerungsmeldung wird allerdings nur einmal, nämlich bei jedem Start des Programms, angezeigt. Wenn Sie das Programm nach Ablauf des 30-tägigen Evaluierungszeitraums weiterverwenden möchten, müssen Sie eine Endbenutzer-Lizenz erwerben, die Sie in Form eines Keycodes erhalten, der in das Dialogfeld "Software-Aktivierung" eingegeben werden muss, um das Produkt zu aktivieren. Sie können Ihre Lizenz online in unserem Web-Shop unter Altova Shop erwerben.

Weitergabe der Software an andere Mitarbeiter in Ihrem Unternehmen zu Testzwecken
Wenn Sie die Evaluierungsversion der Software auch anderen Personen in Ihrem Unternehmen über das Netzwerk zur Verfügung stellen möchten oder wenn Sie sie auf einem PC installieren möchten, der nicht mit dem Internet verbunden ist, sollten Sie nur das Installationsprogramm weitergeben - vorausgesetzt es wurde nicht modifiziert. Jeder, der das von Ihnen zur Verfügung gestellte Installationsprogramm aufruft, muss einen eigenen Evaluierungs-Keycode für 30 Tage anfordern. Nach Ablauf des Testzeitraums, muss eine Lizenz erworben werden, damit das Produkt weiter verwendet werden kann.

Nähere Informationen finden Sie in der Endbenutzer-Lizenzvereinbarung am Ende dieses Handbuchs.

2 Rechte am geistigen Eigentum

Die Software und alle Kopien, deren Anfertigung Ihnen von Altova gestattet ist, sind das geistige Eigentum von Altova und dessen Lieferanten und stehen in deren Besitz. Struktur, Organisation und Code der Software stellen wertvolle Betriebsgeheimnisse und vertrauliche Informationen von Altova und dessen Lieferanten dar. Die Software ist durch gesetzliche Bestimmungen urheberrechtlich geschützt. Diese gesetzlichen Bestimmungen beinhalten (ohne darauf beschränkt zu sein) das Urheberrecht der Vereinigten Staaten, internationale Verträge und das in den Ländern, in denen die Software genutzt wird, geltende Recht. Altova behält sich das Eigentum an allen Patenten, Urheberrechten, Branchengeheimnissen, Warenzeichen und sonstigen geistigen Eigentumsrechten, die hinsichtlich der Software bestehen, vorbehält. Das Eigentumsrecht von Altova erstreckt sich auch auf alle Bilder, Fotografien, Animationen, Videos, Audioaufzeichnungen, Musikstücke, Texte und „Applets“, die Teil der Software sind, und alle dazugehörigen Unterlagen in gedruckter Form. Mitteilungen über geltend gemachte Verstöße gegen geistige Eigentumsrechte sind an den Copyright Agenten von Altova zu richten (nähere Angaben dazu finden Sie auf der Website von Altova).

Altova Software enthält Software-Komponenten Dritter, die ebenfalls urheberrechtlich geschützt sind. Unter anderem sind diese Urheberrechte ausführlich unter http://www.altova.com/legal_3rdparty.html aufgelistet.

Alle anderen Namen oder Warenzeichen sind das Eigentum ihrer jeweiligen Inhaber.

3 Endbenutzer-Lizenzvereinbarung

Altova Endbenutzer Lizenzvertrag für Authentic

DIES IST EIN RECHTSGÜLTIGES DOKUMENT -- BITTE BEWAHREN SIE ES SORGFÄLTIG AUF.

ALTOVA® ENDBENUTZER-LIZENZVERTRAG

FÜR Authentic® Enterprise Software Editions

UND Authentic® Community Software Editions

Lizenzgeber:

Altova GmbH

Rudolfsplatz 13a/9

A-1010 Wien

Österreich

Wichtig – Bitte sorgfältig lesen. Benutzerinformation:

Dieser Altova Endbenutzer Lizenzvertrag für Authentic® ("Lizenzvertrag") regelt Ihr Recht (i) die Authentic Desktop Enterprise Edition Software zu verwenden, (ii) die Authentic Browser Plugin Enterprise Edition Software zu verwenden, zu reproduzieren und zu vertreiben, (iii) die Authentic Desktop Community Edition Software zu verwenden und (iv) die Authentic Browser-Plugin Community Edition Software zu verwenden, zu reproduzieren und zu vertreiben (im folgenden einzeln oder gesamt als „Authentic Software" bezeichnet). Ihre Rechte aus dieser Lizenzvereinbarung hängen von der spezifischen Software Edition ab, über die Sie eine Lizenz erworben haben, da auf einige Editionen verschiedene Rechte und Einschränkungen anwendbar sind, wie im einzelnen im folgenden festgehalten wird. Dieser Endbenutzer-Lizenzvertrag („Lizenzvertrag") ist ein rechtsgültiger Vertrag, der zwischen Ihnen und Altova GmbH („Altova") abgeschlossen wurde. Bevor Sie die von Altova bereitgestellte Software sowie die dazugehörige Dokumentation (u.a. Unterlagen in gedruckter Form, Online-Dateien oder Dokumentation in elektronischer Form („Dokumentation") verwenden, lesen Sie bitte dieses Dokument sorgfältig durch. Indem Sie auf die untenstehenden Schaltflächen „Ich akzeptiere" und „Weiter" klicken, oder indem Sie die Authentic Software installieren oder auf sonstige Weise nutzen, stimmen Sie zu, dass Sie die Bedingungen dieses Lizenzvertrags sowie die Datenschutzbestimmungen („Datenschutzbestimmungen") von Altova, inklusive, jedoch nicht ausschließlich, die nachfolgend festgelegten Garantieausschlüsse, Haftungsbeschränkungen sowie Datennutzungs- und Kündigungsregelungen als rechtsverbindlich anerkennen, auch wenn Sie die Software nicht käuflich erwerben. Sie stimmen zu, dass diese Vereinbarung in demselben Umfang gültig und durchsetzbar ist, wie ein von Ihnen persönlich unterzeichneter Vertrag. Wenn Sie den Bedingungen dieses Lizenzvertrags nicht zustimmen, sind Sie nicht zur Verwendung der Authentic Software berechtigt und müssen alle heruntergeladenen Kopien, die sich in Ihrem Besitz oder unter Ihrer Kontrolle befinden, vernichten. Bitte besuchen Sie unsere Website unter http://www.altova.com/de/authenticeula, um eine Kopie dieses Authentic Lizenzvertrags herunterzuladen und auszudrucken und http://www.altova.com/privacy um die Datenschutzbestimmungen einzusehen.

1. Authentic Desktop Enterprise Edition ("AED") Software Bestimmungen

(a) Lizenzgewährung. Sofern Sie diesem Lizenzvertrag zugestimmt haben, gewährt Ihnen Altova eine nicht ausschließliche und nicht übertragbare (mit Ausnahme der nachfolgenden Bestimmungen) Lizenz, ohne das Recht zur Vergabe von Sublizenzen, zur Installation und Verwendung der AED Software auf einem kompatiblen Einzelplatzrechner oder einer Workstation, wobei die maximale Zahl an Computern, die im

Lizenzumfang enthalten sind, nicht überschritten werden darf. Sie dürfen die AED Software nicht reproduzieren oder vertreiben außer mit ausdrücklicher Genehmigung. Vorbehaltlich der Beschränkungen gemäß Abschnitt 1 (a)(i), dürfen Sie eine Kopie der Software auf mehr als einem Ihrer kompatiblen Einzelplatzrechner oder Workstations installieren und verwenden, wenn Sie eine benutzerdefinierten Lizenz erworben haben. Die maximale Zahl an Computern und/oder Benutzern wird beim Kauf der Software festgelegt und genau angegeben. Während des Evaluierungszeitraums (wie nachfolgend definiert) darf die AED Software nur von einem Benutzer und auf nur einem Einzelplatzrechner oder einer Workstation installiert und genutzt werden. Sie dürfen ein Exemplar der AED Software auf einem Dateiserver innerhalb Ihres internen Netzwerks ausschließlich zum Zweck des Herunterladens und Installieren der AED Software auf anderen Computern innerhalb Ihres Netzwerks bis zur maximalen Anzahl von betrieblich genutzten Computern. Jede sonstige Netzwerknutzung, insbesondere jedoch nicht ausschließlich, die direkte Verwendung oder die Nutzung der AED Software über Befehle, Daten oder Anweisungen von oder an einen Computer, der nicht Teil Ihres internen Netzwerks ist, die Verwendung für Internet- oder Webhosting-Dienste sowie die Verwendung durch einen Benutzer, der nicht unter einer gültigen Lizenz von Altova zur Verwendung dieses Exemplars der Software berechtigt ist, ist unzulässig. Wenn Sie eine Parallellizenz unter den dort vorgesehenen Einschränkungen erworben haben, dürfen Sie ein Exemplar der AED Software auf einem Terminalserver in Ihrem internen Netzwerk für den einzigen und ausschließlichen Zweck installieren, um einzelne Benutzern in Ihrer Organisation von einem anderen Computer innerhalb des gleichen physischen Netzwerks über eine Terminalserver Session den Zugriff auf und die Nutzung der AED Software zu gestatten, vorausgesetzt, dass die Gesamtanzahl der Nutzer, die Zugriff auf die AED Software haben oder diese Nutzen, die maximal zulässige Anzahl nicht übersteigt. Altova gibt hinsichtlich der Performance von Altova-Software in einer Terminalserver-Umgebung keinerlei Zusicherungen oder Gewährleistungen ab und ist eine solche Verwendung von der eingeschränkten Gewährleistung gemäß Abschnitt 3(c) ausgeschlossen; technischer Support hinsichtlich von Problemen, die im Zusammenhang mit der Nutzung in einer solchen Umgebung entstehen, ist nicht verfügbar.

(i) Wenn Sie eine Lizenz der „benutzerdefinierten Version“ der AED Software erworben haben, sind Sie berechtigt, die Software auf bis zu 5 kompatiblen Einzelplatzrechnern oder Workstations zu installieren, auf denen Sie der primäre Nutzer sind. Dadurch können Sie von einem Computer zu einem anderen je nach Bedarf wechseln, vorausgesetzt, dass die Software von Ihnen als benutzerdefinierter Nutzer jeweils nur einmal gleichzeitig verwendet wird. Wenn Sie mehrere benutzerdefinierte Lizenzen erworben haben, erhält jeder individuell definierte Nutzer einen gesonderten Lizenzschlüssel.

(ii) Wenn Sie eine Parallellizenz AED Software-Version erworben haben, können Sie die Software auf mehreren kompatiblen und betrieblich genutzten Computern installieren, jedoch höchstens zehnmal (10-mal) die maximale Zahl der Benutzer, vorausgesetzt, dass nur die maximale Zahl von Benutzern die Software wirklich gleichzeitig nutzen sowie weiters vorausgesetzt, dass die Computer, auf welchen die Software installiert ist, in dasselbe physische Computer Netzwerk eingebunden sind. Die maximale Zahl der Parallelbenutzer wird beim Kauf der AED Software festgelegt. Sofern die Parallelbenutzer Version der Software an mehr als einem Standort oder in mehr als einem Netzwerk verwendet werden soll, sind für jedes physisch getrennte Netzwerk sowie für jeden physisch getrennten Standort separate Parallelbenutzer Lizenzen erforderlich, wobei die oben erwähnten Einschränkungen hinsichtlich der maximalen Zahl der Benutzer und die Anzahl der Nutzer, die die Software benutzen oder Zugang zur Software benötigen, anwendbar sind. Für den Fall, dass ein Computer nicht in dasselbe physische Netzwerk eingebunden ist, ist für diesen eine lokal installierte Benutzerlizenz erforderlich. Be- und Einschränkungen hinsichtlich der Nutzung von Parallelbenutzer Lizenzen auf Heimcomputern sind in diesem Abschnitt geregelt.

(iii) Sie dürfen eine Kopie der AED Software für Sicherungs- und eine Kopie für

 © 2010 Altova GmbH

Archivierungszwecke herstellen, sofern diese nicht auf einem Computer installiert oder verwendet werden. Außerdem müssen bei Kopien zu Sicherungs- und Archivierungszwecken alle im Original enthaltenen urheber- und patentrechtlichen Angaben sowie alle anderen Angaben hinsichtlich geistiger Eigentumsrechte von Ihnen reproduziert werden und in der jeweiligen Kopie der Software unverändert enthalten sein. Sicherungs- und Archivierungskopien dürfen nur dann an Dritte übertragen werden, wenn gleichzeitig eine Übertragung aller Rechte an der Software gemäß dieses Lizenzvertrages erfolgt. Sie als der primäre Nutzer des Computers, auf dem die AED Software installiert ist, dürfen die Software auch auf einem Ihrer privaten Computer zum persönlichen Gebrauch installieren. Eine Kopie der AED Software darf auf privaten Computern bis zur maximalen Anzahl zugelassener Nutzer installiert werden, wenn sichergestellt ist, dass die Software nicht gleichzeitig auf dem privaten Computer und dem primären Computer benutzt wird. Wenn Sie eine Parallel-Nutzer Version der AED Software für den privaten Gebrauch nutzen, dürfen Sie die Software auf jedem kompatiblen Computer installieren, bis zur erlaubten maximalen Anzahl von Nutzern.

(b) Freischaltcodes. Bevor Sie die Software kaufen bzw. im Rahmen Ihrer Registrierung für den Evaluierungszeitraum von dreißig (30) Tagen erhalten Sie einen Evaluierungs-Freischaltcode. Wenn Sie sich in der Folge dazu entscheiden, die AED Softwarelizenz von Altova GmbH oder einem autorisierten Vertriebshändler zu kaufen, erhalten Sie einen uneingeschränkten Freischaltcode. Mit diesem können Sie die Software nach dem Evaluierungszeitraum aktivieren. Ohne ausdrückliche schriftliche Genehmigung von Altova dürfen Sie an Freischaltcodes keine Lizenzen einräumen und Freischaltcodes nicht reproduzieren oder vertreiben.

(c) Übertragungsbeschränkung. Sie dürfen alle Ihre Rechte zur Verwendung der AED Software an eine andere natürliche oder juristische Person unter der Voraussetzung übertragen, dass: (i) Sie den vorliegenden Lizenzvertrag, die AED Software und sonstige Soft- oder Hardware, die mit der AED Software, die Gegenstand dieses Lizenzvertrags ist, geliefert, verpackt oder auf dieser vorinstalliert ist, einschließlich aller Kopien, Updates und früherer Versionen sowie aller Exemplare von Fontsoftware, die in andere Formate konvertiert wurde, an die betreffende Person übertragen; (ii) Sie keine Kopien, einschließlich Sicherungskopien und sonstiger, auf einem Computer gespeicherter Kopien, zurückbehalten; (iii) der Empfänger von Altova einen persönlichen Freischaltcode erhält; und (iv) der Empfänger die Bestimmungen dieses Lizenzvertrags sowie alle anderen Bestimmungen akzeptiert, zu denen Sie die AED Softwarelizenz rechtmäßig erworben haben.

(d) Anwendbare Lizenzvertragsbestimmungen. Die Abschnitte 1, 3 und 7 des Lizenzvertrages sind auf die AED Software anzuwenden.

2. <u>Authentic Browser-Plugin Enterprise Edition ("ABE") Software Bestimmungen</u>

(a) Lizenzgewährung und Bestimmungen. Sofern Sie diesem Lizenzvertrag zugestimmt haben, gewährt Ihnen Altova eine nicht ausschließliche und nicht übertragbare Lizenz, ohne das Recht zur Vergabe von Sublizenzen, zur Installation und Verwendung der ABE Software auf einer per Server Basis für einen Zeitraum von zwölf (12) Monaten, beginnend mit dem Datum Ihres Lizenzerwerbs und endend mit dem Datum, das zwölf (12) Monate danach liegt (der "ABE Lizenz Zeitraum"). Altova gewährt Ihnen auch eine nicht ausschließliche und nicht übertragbare, eingeschränkte weltweite Lizenz, ohne das Recht zur Vergabe von Sublizenzen, die Software zu benutzen, um Web Pages, Web Applikationen oder Anwendungen, welche ABE Software beinhalten zu entwickeln, ABE Software auf Ihrer Website oder Ihrem Server zu reproduzieren und ABE Software von Ihrer Website oder Ihrem Server aus über ein Computer Netzwerk zu vertreiben, aber nur in ihrer ausführbaren Object Code Form, und nur an Endverbraucher, ausschließlich für den eingeschränkten Zweck, diesen zu ermöglichen, während des ABE Lizenz-Zeitraums, XML Dateien zu sehen, zu übertragen und/oder zu bearbeiten. Wenn Sie nach Beendigung des ABE Lizenz-Zeitraums weiterhin die ABE Software verwenden, und/oder wiedergeben und/oder vertreiben wollen, müssen Sie eine neue Authentic Browser-Plugin Enterprise Edition erwerben. Wenn Sie eine ABE Software Lizenz unter den Bedingungen des Lizenzvertrages erworben haben, sind Support und Wartung (oder SMP wie im folgenden weiter ausgeführt) für die Software als Teil des Lizenzkaufs inbegriffen und sind sie berechtigt, die im folgenden ausgeführten Vorteile für die Dauer des ABE Lizenz-Zeitraums, welcher gleichzeitig mit dem Support Zeitraum läuft in Anspruch zu nehmen.

Anders als bei anderen Altova Software Produkten können Sie SMP für ABE Software nicht erneuern und müssen, nach Beendigung des Lizenz-Zeitraums und des Support Zeitraums, eine neue ABE Software Lizenz erwerben, falls Sie die Software weiter benutzen, reproduzieren oder vertreiben wollen.

(b) Freischalt Codes. Vor Ihrem Kauf haben Sie die Möglichkeit, sich einen dreißig (30) Tage gültigen Evaluierungsfreischaltcode zusenden zu lassen. **Wenn Sie sich entschließen die ABE Software Lizenzen entweder von der Altova GmbH oder einem autorisierten Vertriebshändler zu erwerben, erhalten Sie einen Kauf Freischaltcode. Der Kauf Freischaltcode ermöglicht es Ihnen die Software während des ABE Lizenz-Zeitraums zu aktivieren. Ohne ausdrückliche schriftliche Genehmigung von Altova dürfen Sie an Freischaltcodes keine Lizenzen einräumen, und Freischaltcodes nicht reproduzieren, oder weitergeben.**

(c) ABE Software-spezifische Einschränkungen. Zusätzlich zu den Einschränkungen und Verpflichtungen, die sich aus anderen Abschnitten dieses Lizenzvertrages welche auf ABE Software anzuwenden sind, ergeben, ist ihre eingeschränkte Lizenz, die ABE Software zu vertreiben wie sie oben festgehalten wurde, folgenden Einschränkungen unterworfen: (i) ABE Software darf nur lizenziert jedoch nicht verkauft werden, und (ii) Sie müssen die von Altova zur Verfügung gestellte ABE Software verwenden, wiedergeben oder vertreiben, **SO WIE SIE IST** und dürfen Altovas Lizenzvertrag (welcher während des Installationsprozesses aufscheint und von dem Endbenutzer akzeptiert werden muss, um die Möglichkeit zu haben diese Software oder andere Dateien zu installieren oder zu benutzen) nicht beschädigen, verändern oder entfernen.

(d) Anwendbare Lizenzvertragsbestimmungen. Die in den Abschnitten 2, 3 und 7 festgehaltenen Bestimmungen sind auf ABE Software anzuwenden.

3. Authentic Enterprise Editions (AED und ABE) Software Bestimmungen

Die Bestimmungen welche in Abschnitt 3 festgehalten werden sind auf AED und ABE Software Lizenzen anzuwenden, und gelten zusätzlich zu den spezifischen Bestimmungen welche auf diese Software Lizenzen anzuwenden sind.

(a) Upgrades und Updates. Wenn die von Ihnen in Lizenz erworbene Software ein Upgrade oder Update ist, beendet das Upgrade oder Update das sie zuletzt heruntergeladen oder installiert haben die Lizenz für die zuvor lizenzierte Kopie der AED oder ABE Software in dem Ausmaß in dem sie ersetzt wird. Das Update oder Upgrade und die damit verbundenen Lizenz Freischaltcodes, soweit anwendbar, stellen insofern nicht die Gewährung einer zweiten Lizenz für die Software dar, als es untersagt ist die upgegradete oder upgedatete Kopie zusätzlich zu derjenigen Kopie der Software zu benutzen welche dadurch ersetzt und deren Lizenz dadurch beendet wird.

(b) Support und Wartung. Altova bietet Support und Wartungs-Pakete ("SMP") für die AED und ABE Software Produkt Editionen an, an welchen sie eine Lizenz erworben haben. Der Support Zeitraum, wie nachfolgend definiert, umfasst ein solches SMP und wird bestimmt wenn Sie sich entschließen ein SMP zu erwerben. Im Falle Ihrer ABE Software Lizenz sind Zwölf Monate SMP inbegriffen, welche zeitgleich mit dem ABE Lizenz-Zeitraum ablaufen. Ihre Rechte im Zusammenhang mit Support und Wartung sowie Ihre Möglichkeit ein Upgrade zu erhalten hängen davon ab, ob Sie sich entschieden haben ein SMP zu erwerben bzw. welche Art von SMP sie erworben haben:

(i) Wenn Sie kein SMP erwerben, erhalten Sie beim Kauf lediglich die Software, danach jedoch keinerlei Wartungsreleases oder Updates. Obwohl die Möglichkeit besteht, dass Altova Ihnen in Einzelfällen Wartungsreleases als kostenlose Zusatzleistung zur Verfügung stellt, sind in diesen Releases keine neuen Produktfeatures, die über das beim Kauf der Software bestehende Maß hinausgehen, beinhaltet. In jedem Fall erhalten Sie dreißig (30) Tage lang ab Kauf der Software (der „Supportzeitraum" im Sinne dieses Absatzes (b)) kostenlosen technischen Support von Altova. Außerdem kann Altova in Einzelfällen auch während des dreißig (30) Tage dauernden Evaluierungszeitraums technischen Support als kostenlose Zusatzleistung zur Verfügung stellen. Technischen Support erhalten Sie ausschließlich über ein webbasiertes Supportformular, wobei es keine

 © 2010 Altova GmbH

garantierte Reaktionszeit gibt.

(ii) Mit dem Erwerb eines SMP haben Sie während des dafür geltenden Supportzeitraums Anspruch auf die von Ihnen erworbene Produktversion sowie auf alle Wartungsreleases und Updates für diese Produktversion, die während des für Sie geltenden Supportzeitraumsfreigegeben werden. Während des Supportzeitraums Ihres SMP erhalten Sie auch Upgrades auf die entsprechende Produktversion der AED oder ABE Software, mit der auf eine höhere Version der Software, für die Sie die Lizenz erworben haben, gewechselt wird und die während Ihres Supportzeitraums freigegeben werden. Die einzelnen Upgradeeditionen, auf die Sie innerhalb Ihres Supportzeitraums Anspruch haben, sind in dem von Ihnen erworbenen SMP im Detail angeführt. Software, die als gesondertes Produkt eingeführt wird, ist nicht im SMP enthalten. Wartungsreleases, Updates und Upgrades können neue Features enthalten, dies muss aber nicht der Fall sein. Darüber hinaus erhalten Sie während des Supportzeitraums bevorzugten technischen Support von Altova, und zwar ausschließlich über ein webbasiertes Supportformular. Altova wird alle wirtschaftlich vertretbaren Anstrengungen unternehmen, um alle Anfragen per E-Mail innerhalb von achtundvierzig (48) Stunden während der Geschäftszeiten (Montag bis Freitag, 8.00 bis 22.00 Uhr UTC, ausgenommen Feiertage in Österreich und den USA) zu beantworten. Außerdem wird Altova sich in angemessenem Umfang darum bemühen, Workarounds für Fehler, die bei der Software aufgetreten sind, zur Verfügung zu stellen.

(iii) Während des Supportzeitraums können Sie Altova Fehler oder Defekte in der Software melden. Wenn nach Altovas Einschätzung ein reproduzierbarer schwerwiegender Fehler vorliegt, der die Verwendung und Funktionalität der AED oder ABE Software erheblich beeinträchtigt, wird Altova wirtschaftlich vertretbaren Anstrengungen unternehmen um Korrekturen oder provisorische Lösungen anzubieten, die in zukünftigen Updates oder Wartungsreleases enthalten sind. Diese Updates oder Wartungsreleases werden von Altova von Zeit zu Zeit zur Verfügung gestellt.

Es steht im Ermessen von Altova, einen schriftlichen Nachweis über von Ihnen festgestellte Fehler oder Funktionsstörungen zu verlangen oder Beispieldateien anzufordern, aus denen das aufgetretene Softwareproblem hervorgeht. In einem solchen Fall müssen Sie die angeforderten Nachweise oder Dateien, aus denen ausreichend detailliert hervorgeht, in welchen Aspekten Fehler bei der AED oder ABE Software auftreten, so schnell wie möglich per E-Mail, Fax oder Expresspost mit Zustellung am nächsten Tag an Altova übermitteln. Bei der Diagnose oder Analyse von Fehlern haben Sie in zumutbarem Rahmen mit Altova zu kooperieren. Fehlerbehebungen können in Wartungsreleases, Updates oder neuen Hauptversionen der Software enthalten sein. Altova ist nicht verpflichtet, unwesentliche Fehler, d.h. Fehler, die die Benutzung der Software nicht wesentlich beeinträchtigen, zu beheben. Unabhängig davon, ob Sie das Support- und Wartungspaket erworben haben, wird technischer Support ausschließlich für Fragen oder Probleme angeboten, die sich unmittelbar aus oder in Zusammenhang mit der AED oder ABE Software ergeben. Sie erhalten von Altova unter keinen Umständen Beratung, Unterstützung oder Hilfestellung, die allgemeiner Natur ist und nicht in unmittelbarem Zusammenhang mit der Software steht.

(iv) für ein Update der AED oder ABE Software kann es notwendig sein, Software, welche nicht von diesem Lizenzvertrag abgedeckt wird, vor der Installation zu aktualisieren (updaten). Updates des Betriebssystems und von Anwendungssoftware, welche nicht ausdrücklich von diesem Lizenzvertrag abgedeckt ist, liegen in Ihrer Verantwortung und werden nicht von Altova aufgrund dieses Lizenzvertrages zur Verfügung gestellt. Altovas Verpflichtungen unter diesem Abschnitt sind abhängig von Ihrem angemessenen Gebrauch der Software und Ihrer Einhaltung der Bestimmungen dieses Lizenzvertrages zu allen Zeitpunkten. Altova ist keinesfalls verpflichtet den oben genannten technischen Support zur Verfügung zu stellen, wenn nach Altovas Ansicht das Gebrechen an der AED oder ABE Software durch folgende Umstände ausgelöst wurde: (aa) Schaden, der an der Software entstanden ist, durch den Transport an einen anderen Ort oder Computer; (bb) Änderungen, Modifikationen oder Versuche die Software zu ändern, ohne Altovas schriftliche Zustimmung; (cc) Ursachen die nicht in der Software begründet liegen, wie Naturkatastrophen, der Ausfall oder das Schwanken des elektrischen Stroms, oder ein Ausfall von Teilen des Computers; (dd) Ihre Unterlassung, die Software auf dem von Altova festgelegten Release Level zu halten; oder (ee) die Benutzung der Software gemeinsam mit anderer Software ohne Altovas vorherige schriftliche Zustimmung. Es liegt in Ihrer alleinigen

Verantwortung: (x) alle von Altova spezifizierten Benutzungs- und Fehlerbehebungsprozeduren einzuhalten und danach Altova umgehend über alle AED und ABE Software Fehlfunktionen zu benachrichtigen und Altova sämtliche Informationen darüber zur Verfügung zu stellen; (y) für die Sicherheit Ihrer vertraulichen Informationen zu sorgen; und (z) Backup Systeme und Prozeduren einzuführen und aufrechtzuerhalten, welche notwendig sind um verlorene oder veränderte Dateien, Daten oder Programme wiederherzustellen.

(c) EINGESCHRÄNKTE GARANTIE. Altova garantiert der Person/Rechtspersönlichkeit, die ursprünglich die Lizenz für die Verwendung der AED oder ABE Software gemäß den Bestimmungen dieses Lizenzvertrages erworben hat, dass (i) die Software über einen Zeitraum von neunzig (90) Tagen nach Erhalt im Wesentlichen in Übereinstimmung mit der dazugehörigen Dokumentation funktioniert, und (ii) die von Altova zur Verfügung gestellten Supportleistungen im Wesentlichen auf die in Abschnitt 3(b) dieses Vertrags niedergelegte Weise erfolgen. In einigen Ländern sind Beschränkungen über die Dauer einer stillschweigenden Garantie nicht zulässig, so dass die obigen Beschränkungen und Ausschlüsse eventuell für Sie nicht zutreffen. Im größtmöglichen rechtlich zulässigen Maß sind stillschweigende Garantien in Bezug auf die AED oder ABE Software (sofern solche existieren) auf neunzig (90) Tage beschränkt. Die gesamte Haftung von Altova und dessen Lieferanten, sowie Ihre einzigen Garantieansprüche, sind nach dem Ermessen von Altova auf eine der beiden folgenden Optionen beschränkt: (i) Erstattung des Kaufpreises (wenn zutreffend), oder (ii) Reparatur oder Austausch der AED oder ABE Software, die unter die eingeschränkte Garantie von Altova fällt und die unter Vorlage einer Kopie des Kaufbelegs bei Altova reklamiert wird. Diese eingeschränkte Garantie gilt nicht, wenn die Funktionalität der AED oder ABE Software durch ein Versehen, durch Missbrauch, falsche Anwendung, Trojaner, Viren oder einen sonstigen schädlichen externen Code beeinträchtigt wurde. Die Garantie für jede Ersatzsoftware erstreckt sich auf die Restdauer des ursprünglichen Garantiezeitraums oder auf dreißig (30) Tage, je nachdem, welcher Zeitraum länger ist. Die eingeschränkte Garantie gilt nicht für Test-Software. MIT AUSNAHME DER VORSTEHEND ANGEFÜHRTEN EINGESCHRÄNKTEN GARANTIE UND DEN DIESBEZÜGLICHEN GARANTIEANSPRÜCHEN BESTEHEN SEITENS ALTOVA ODER DESSEN LIEFERANTEN KEINERLEI WEITEREN GARANTIEVERPFLICHTUNGEN. ALTOVA UND DESSEN LIEFERANTEN ÜBERNEHMEN KEINE GEWÄHR FÜR DIE ANWENDUNG ODER DIE ERGEBNISSE AUS DER NUTZUNG DER SOFTWARE. MIT AUSNAHME DER VORSTEHEND ANGEFÜHRTEN EINGESCHRÄNKTEN GARANTIE SOWIE IM HINBLICK AUF ALLE ANDEREN GEWÄHRLEISTUNGEN, BEDINGUNGEN, ZUSICHERUNGEN ODER ANSPRÜCHE, DIE NACH DER FÜR IHR LAND GELTENDEN RECHTSORDNUNG NICHT AUSGESCHLOSSEN ODER EINGESCHRÄNKT WERDEN KÖNNEN, SCHLIESSEN ALTOVA UND DESSEN LIEFERANTEN ALLE ANDEREN AUSDRÜCKLICHEN ODER STILLSCHWEIGENDEN GEWÄHRLEISTUNGEN, BEDINGUNGEN, ZUSICHERUNGEN UND ANSPRÜCHE AUS, DIE SICH AUS DEM GESETZ, DER RECHTSPRAXIS, EINEM GEWOHNHEITSRECHT, EINEM HANDELSBRAUCH ODER AUS SONSTIGEN GRÜNDEN ERGEBEN. ALTOVA UND DESSEN LIEFERANTEN SCHLIESSEN IM GRÖSSTMÖGLICHEN RECHTLICH ZULÄSSIGEN UMFANG ALLE ANDEREN AUSDRÜCKLICHEN UND STILLSCHWEIGENDEN GARANTIEN UND GEWÄHRLEISTUNGEN AUS. DIES BEINHALTET UNTER ANDEREM GARANTIEN IM HINBLICK AUF MARKTGÄNGIGKEIT, EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, ZUFRIEDENSTELLENDE QUALITÄT, INFORMATIONSGEHALT ODER –RICHTIGKEIT, UNGESTÖRTE NUTZUNG, EIGENTUMSRECHT UND NICHTVERLETZUNG VON RECHTEN DRITTER UND DIE ERBRINGUNG ODER NICHTERBRINGUNG VON SUPPORTLEISTUNGEN IN BEZUG AUF DIE SOFTWARE. MIT DIESER EINGESCHRÄNKTEN GARANTIE WERDEN IHNEN BESTIMMTE RECHTE EINGERÄUMT. UNTER UMSTÄNDEN BESITZEN SIE NOCH ANDERE RECHTE, DIE JE NACH LAND/RECHTSORDNUNG UNTERSCHIEDLICH SEIN KÖNNEN.

(d) Haftungsbeschränkung und Ansprüche in Zusammenhang mit Urheberrechtsverletzungen. SOWEIT DIE ANWENDBAREN GESETZE DIES ZULASSEN, HAFTEN ALTOVA ODER DESSEN LIEFERANTEN, AUCH WENN EINE IM RAHMEN DER GARANTIE DURCHGEFÜHRTE ABHILFEMASSNAHME IHREN WESENTLICHEN ZWECK NICHT ERFÜLLT, IN KEINEM FALL FÜR KONKRETE, ZUFÄLLIG ENTSTANDENE,

© 2010 Altova GmbH

MITTELBARE, UNMITTELBARE ODER FOLGESCHÄDEN JEGLICHER ART (INSBESONDERE SCHÄDEN AUS ENTGANGENEM GEWINN, GESCHÄFTSUNTERBRECHUNGEN, VERLUST VON GESCHÄFTSINFORMATIONEN ODER ANDEREN FINANZIELLEN VERLUSTEN), DIE DURCH DIE NUTZUNG ODER DIE UNMÖGLICHKEIT DER NUTZUNG DER SOFTWARE ODER DIE NICHTERBRINGUNG VON SUPPORTLEISTUNGEN ENTSTANDEN SIND, SELBST WENN ALTOVA AUF DIE MÖGLICHKEIT EINES SOLCHEN SCHADENS HINGEWIESEN WURDE. IN JEDEM FALL IST DIE GESAMTHAFTUNG VON ALTOVA GEMÄSS DIESEM LIZENZVERTRAG AUF DIE HÖHE DES FÜR DAS SOFTWAREPRODUKT ENTRICHTETEN BETRAGES BESCHRÄNKT. Da es in einigen Ländern und Rechtsordnungen nicht zulässig ist, die Haftung für Schäden auszuschließen oder zu beschränken, gilt die obige Einschränkung unter Umständen nicht für Sie. In solchen Ländern und Rechtsordnungen gilt die Beschränkung der Haftung durch Altova im größtmöglichen rechtlich zulässigen Umfang, und der Ausschluss bzw. die Beschränkung der in diesem Lizenzvertrag beinhalteten Garantien und Haftungen hat keine Auswirkung auf die Konsumentenschutzrechte von Personen, die Waren auf andere Weise als im Zuge ihrer geschäftlichen Tätigkeit erwerben. Der vorstehende Haftungssauschluss/die vorstehende Haftungsbeschränkung sind wesentliche Bestandteile des zwischen Ihnen und Altova abgeschlossenen Lizenzvertrags. Altova wird Sie gegenüber allen Forderungen, Prozessen oder Verfahren schad- und klaglos halten bzw. alle Ansprüche, Prozesse oder Verfahren beilegen, die Dritte mit dem Argument gegen Sie erheben, dass der Inhalt der AED oder ABE Software gegen ein Urheberrecht verstößt oder ein geistiges oder sonstiges Eigentumsrecht verletzt, das durch das Recht der Vereinigten Staaten oder der Europäischen Union geschützt ist (insgesamt als „Ansprüche“ bezeichnet). Dies erfolgt – soweit nicht ausdrücklich etwas Anderes festgelegt ist – jedoch nur insoweit, als der betreffende Anspruch sich direkt aus der Verwendung der Software ergibt und nach Maßgabe der in Abschnitt 3(d) dieses Vertrags festgelegten Beschränkungen. Altova ist von jedem Anspruch innerhalb von zehn (10) Geschäftstagen, nachdem Sie erstmals davon benachrichtigt wurden, in Kenntnis zu setzen. Außerdem haben Sie mit Altova in angemessenem Umfang bei der Abwehr solcher Ansprüche zu kooperieren und Altova dabei zu unterstützen, ohne dass Sie dafür Kosten geltend machen können. Das Recht auf Entscheidungen in Bezug auf solche Ansprüche liegt allein bei Altova (dies beinhaltet auch, ohne darauf beschränkt zu sein, die Auswahl der Rechtsberater und das Recht, für Sie einen Vergleich zu den von Altova für zweckmäßig erachteten Bedingungen einzugehen). Sie können auf eigene Kosten einen Rechtsberater hinzuziehen und an den Verfahrens- oder Vergleichsverhandlungen teilnehmen. Altova kommt bis zu einer Höhe von insgesamt maximal dem Kaufpreis der AED oder ABE Software für die Schäden, Kosten und Anwaltsgebühren auf, zu deren Bezahlung Sie in Zusammenhang mit solchen Ansprüchen verpflichtet werden (oder die Sie aufgrund eines Vergleichs zu entrichten haben), soweit diese nicht von einer Versicherung oder dritten Partei übernommen werden. Ist oder wird die AED oder ABE Software Gegenstand von aufgrund einer Urheberrechtsverletzung vorgebrachten Ansprüchen, oder wird ihre Verwendung untersagt, oder ist es nach Ansicht des Rechtsberaters von Altova wahrscheinlich, dass ein solcher Umstand eintritt, so wird Altova versuchen, eine Beilegung herbeizuführen, indem alle wirtschaftlich vertretbaren Anstrengungen unternommen werden, um die Software zu modifizieren oder eine Lizenz für die weitere Verwendung der Software zu erwerben. Wenn es nach Ansicht des Rechtsberaters von Altova nicht möglich ist, den bevorstehenden oder bereits vorgebrachten Anspruch bzw. die Verfügung, mit der die Verwendung der Software untersagt wurde, durch angemessene Abänderung oder den Erwerb einer Lizenz beizulegen, so kann Altova diesen Lizenzvertrag ohne negative Konsequenzen für Altova beenden und Ihnen anteilig alle bereits an Altova entrichteten Gebühren zurückerstatten. MIT AUSNAHME DER VORSTEHEND ANGEFÜHRTEN BESTIMMUNGEN OBLIEGEN ALTOVA KEINERLEI HAFTUNGSVERPFLICHTUNGEN FÜR ANSPRÜCHE IN ZUSAMMENHANG MIT URHEBERRECHTSVERLETZUNGEN. Diese Haftungsverpflichtung gilt nicht für Urheberrechtsverletzungen, die ausschließlich auf vom Kunden eingebrachte Elemente zurückzuführen sind.

(e) Test Software. Dieser Abschnitt findet Anwendung auf alle Test Kopien der AED oder ABE Software (“Test Software”) und behält seine Gültigkeit bis Sie eine Lizenz erwerben. DIE TESTSOFTWARE WIRD IHNEN WIE BESEHEN UND DAHER OHNE GEWÄHR FÜR DEREN VERWENDUNG ODER LEISTUNGSMERKMALE ZUR VERFÜGUNG GESTELLT, UND ALTOVA ÜBERNIMMT DIESBEZÜGLICH KEINERLEI AUSDRÜCKLICHEN ODER STILLSCHWEIGENDEN GEWÄHRLEISTUNGS- ODER HAFTUNGSVERPFLICHTUNGEN.

SOFERN UND SOWEIT FÜR TESTSOFTWARE KEIN HAFTUNGSSAUSSCHLUSS, SONDERN NUR EINE HAFTUNGSBESCHRÄNKUNG GESETZLICH ZULÄSSIG IST, BESCHRÄNKT SICH DIE HAFTUNG VON ALTOVA UND DESSEN LIEFERANTEN AUF INSGESAMT FÜNFZIG US-DOLLAR (50 USD). Wenn die Testversion mit einer Time-out-Funktion ausgestattet ist, kann sie nach Ablauf des Evaluierungszeitraums nicht mehr verwendet werden. Der Zugriff auf sämtliche Dateien, welche unter Benützung der Test Software errichtet wurden, erfolgt auf eigene Gefahr.

4. Authentic Desktop Community Edition ("ACDE") Software Bestimmungen

(a) Lizenzgewährung und Freischaltcode. Sofern Sie diesem Lizenzvertrag zugestimmt haben, gewährt Ihnen Altova eine nicht ausschließliche und nicht übertragbare eingeschränkte Lizenz, ohne dem Recht Sublizenzen einzuräumen, eine Kopie der ACDE Software auf Ihrem kompatiblen Einzelplatzrechner oder Ihrer kompatiblen Workstation zu installieren und zu verwenden, um XML Dateien ausschließlich in Verbindung mit STYLEVISION® Power Stylesheets, wie im folgenden ausgeführt, zu betrachten, zu vertreiben, zu teilen oder zu verändern.

Es ist Ihnen untersagt den Freischalt Code zu re-lizenzieren, zu reproduzieren, oder weiterzugeben, außer mit der ausdrücklichen schriftlichen Erlaubnis von Altova. Sie dürfen eine Kopie der ACDE Software für Sicherungs- und eine Kopie für Archivierungszwecke herstellen, sofern diese nicht auf einem Computer installiert oder verwendet werden. Außerdem müssen bei Kopien zu Sicherungs- und Archivierungszwecken alle im Original enthaltenen urheber- und patentrechtlichen Angaben sowie alle anderen Angaben hinsichtlich geistiger Eigentumsrechte von Ihnen reproduziert werden und in der jeweiligen Kopie der Software unverändert enthalten sein. Es ist Ihnen gestattet eine Kopie eines solchen Setup Programms für ACDE Software auf einem Datei Server in Ihrem internen Netzwerk zu installieren für den einzigen und ausschließlichen Zweck die ACDE Software zu installieren (auf einer unbegrenzten Anzahl von Client Computern innerhalb Ihres privaten Netzwerks). Keinerlei andere Server oder Netzwerk Nutzung der ACDE Software ist gestattet, einschließlich aber nicht ausschließlich der Benutzung der ACDE Software (i) entweder direkt oder durch Befehle, Daten oder Anweisungen von oder an andere Computer oder (ii) für interne Netzwerk-, Internet- oder Web-hosting Dienste.

(b) Vertrieb. Zum Zeitpunkt Ihrer Annahme dieses Lizenzvertrages als Teil ihrer Benutzung der ACDE Software und unter der Voraussetzung ihrer fortgesetzten Einhaltung dessen Bestimmungen, räumt Altova hiermit den Nutzern von ACDE Software eine nicht ausschließliche, nicht übertragbare, beschränkte Lizenz ein, ohne das Recht Sublizenzen zu vergeben, das Setupprogramm der ACDE Software zu reproduzieren und das Setup Programm für die ACDE Software in ausführbarer Form in der im folgenden beschriebenen Art und Weise an Endbenutzer zu verteilen. Sie dürfen das Setup Programm für die ACDE Software an jedweden Dritten elektronisch oder als Download von Ihrer Website oder auf physikalischen Datenträgern wie CD-ROMS oder Disketten als Teil von oder in Verbindung mit Produkten, welche Sie entwickelt haben, weitergeben.

(c) ACDE Software Spezifische Einschränkungen. Zusätzlich zu den Einschränkungen und Verpflichtungen welche sich aus anderen Abschnitten dieses Lizenzvertrags ergeben, ist die Ihnen eingeräumte Lizenz das Setup Programm für ACDE zu vertreiben sämtlichen der folgenden Einschränkungen unterworfen: (i) ACDE Software kann nur lizenziert und darf nicht verkauft werden (ii) es ist Ihnen nicht gestattet die ACDE Software als Stand-Alone Produkt zugänglich zu machen und, wenn sie als Teil eines Produkt-Bundles vertrieben wird, ist es Ihnen erlaubt hiefür einen Preis festzusetzen, sofern dieses Produkt-Bundle zu dem selben oder einem niedrigeren Preis als jenem, zu dem ein gleichwertiges Produkt-Bundle,, welches die ACDE Software nicht enthält, lizenziert wird, (iii) Sie müssen das von Altova zur Verfügung gestellte Setup Programm für die ACDE Software unverändert benutzen und dürfen Altovas Lizenzvertrag (welcher während des Installationsprozesses aufscheint und vom Endbenutzer akzeptiert werden muss um die ACDE Software installieren oder benutzen zu können) oder andere Dateien nicht beschädigen verändern oder entfernen, (iv) es ist Ihnen weiters untersagt, die ACDE Software mit Ihrem Produkt auf solche Weise zu verbinden, dass es Stylevision Power Stylesheets modifiziert oder erstellt.

(d) Anwendbare Lizenzvertrags Bestimmungen. Die in den Abschnitten 4, 6 und 7 festgehaltenen

Bestimmungen sind auf ACDE Software anzuwenden.

5. Allgemeine Authentic Browser-Plugin Community Edition ("ACBE") Software Bestimmungen

(a) Lizenzgewährung und Vertrieb. Sofern Sie diesem Lizenzvertrag zugestimmt haben, gewährt Ihnen Altova eine nicht ausschließliche und nicht übertragbare eingeschränkte Lizenz, ohne das Recht Sublizenzen anzuräumen, Web Pages, Web Anwendungen, oder Anwendungen welche ACBE Software beinhalten, zu benutzen und zu entwickeln, die ACBE Software zu reproduzieren und die ACBE Software in ausführbarer Form in der Art und Weise, welche im folgenden festgelegt wird, an Endbenutzer zu vertreiben um XML Dateien ausschließlich in Verbindung mit STYLEVISION® Power Stylesheets, wie im folgenden ausgeführt, zu betrachten, zu vertreiben, zu teilen oder zu verändern. Es ist Ihnen gestattet eine Kopie der ACBE Software auf einem Web Server in Ihrem internen Netzwerk zu installieren für den einzigen und ausschließlichen Zweck, die ACDE Software zu installieren (auf einer unbegrenzten Anzahl von Client Computern innerhalb Ihres privaten Netzwerks).

Sie dürfen das Setup Programm für die ACBE Software an jedweden Dritten elektronisch oder als Download von Ihrer Website oder auf physischen Datenträgern wie CD-ROMS oder Disketten als Teil von oder in Verbindung mit Produkten, welche Sie entwickelt haben, weitergeben.

(b) ACBE Software spezifische Einschränkungen. Zusätzlich zu den Einschränkungen und Verpflichtungen, welche sich aus anderen Abschnitten dieses Lizenzvertrags ergeben, ist die Ihnen eingeräumte Lizenz, das Setup Programm für ACBE zu vertreiben, sämtlichen der folgenden Einschränkungen unterworfen: (i) ACBE Software kann nur lizenziert und nicht verkauft werden (ii) es ist Ihnen nicht gestattet die ACBE Software als Stand-Alone Produkt zugänglich zu machen und, wenn diese als Teil eines Produkt-Bundles vertrieben wird, ist es Ihnen erlaubt hiefür einen Preis festzusetzen, sofern dieses Produkt-Bundle zu dem selben oder einem niedrigeren Preis als jenem, zu dem ein gleichwertiges Produkt-Bundle, welches die ACBE Software nicht enthält, lizenziert wird, (iii) Sie müssen das von Altova zur Verfügung gestellte Setup Programm für die ACBE Software unverändert benutzen und dürfen Altovas Lizenzvertrag (welcher während des Installationsprozesses aufscheint und vom Endbenutzer akzeptiert werden muss, um die ACBE Software installieren oder benutzen zu können) oder andere Dateien nicht beschädigen verändern oder entfernen; (iv) andere Altova Produkte können nicht unter diesem Lizenzvertrag vertrieben werden; (v) es ist Ihnen weiters untersagt die ACBE Software mit Ihrem Produkt so zu verbinden, dass es Stylevision Power Stylesheets modifiziert oder erstellt.

(c) Anwendbare Lizenzvertragsbestimmungen. Die in den Abschnitten 5, 6 und 7 festgehaltenen Bestimmungen sind auf ACBE Software anzuwenden.

6. Authentic Community Editions (ACDE und ACBE) Software Bestimmungen

Die in Abschnitt 6 festgehaltenen Bestimmungen sind anwendbar auf ACDE und ACBE Software Lizenzen und gelten zusätzlich zu den spezifischen Bestimmungen welche auf diese Software Lizenzen anwendbar sind.

(a) Benutzungsbeschränkung. Die ACDE und ACBE Software werden von Altova lizenziert und vertrieben und dienen der Betrachtung, dem Vertrieb, der Teilung und Bearbeitung von XML Dateien ausschließlich in Verbindung mit STYLEVISION Power Stylesheets, definiert als .sps Dateien, bei welchen es sich um von Altova oder von Kunden durch Altovas STYLEVISION Produkt erstellte Template Dateien handelt. Es ist Ihnen nicht gestattet die ACDE oder ACBE Software in oder mit (i) einem/mehreren STYLEVISION Power Stylesheet(s), welche nicht in Übereinstimmung mit dem Altova Software Lizenzvertrag, welcher unter http://www.altova.com/eula einsehbar ist, erstellt wurden oder (ii) andere Software oder Verbesserungen welche die Inter Application Communication (IAC) verwenden um auf Programmebene ein Interface mit Authentic Software herzustellen um zusätzliche Funktionen zu ermöglichen welche normalerweise bei Authentic nicht möglich wäre oder welche Funktionen ermöglichen die mit Altova Programmen konkurrieren, zu integrieren oder zu nutzen.

(b) Garantie, Haftungseinschränkungen. DIE ACDE ODER ACBE SOFTWARE WIRD IHNEN GRATIS UND WIE BESEHEN („AS-IS") ZUR VERFÜGUNG GESTELLT. ALTOVA GIBT

KEINERLEI GEWÄHRLEISTUNG FÜR DIE ACDE ODER ACBE SOFTWARE AB. ALTOVA UND IHRE ZULIEFERER GEBEN SOWOHL AUSDRÜCKLICH ALS AUCH STILLSCHWEIGEND UND AUF JEDE ANDERE WEISE IM GESETZLICH GRÖßTMÖGLICHEN AUSMAß KEINERLEI GEWÄHRLEISTUNGEN ODER ZUSICHERUNGEN, INSBESONDERE JEDOCH NICHT AUSSCHLIEßLICH. FÜR ALLE IMPLIZITEN GEWÄHRLEISTUNGEN DER EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, FÜR MARKTGÄNGIGKEIT, FÜR BEFRIEDIGENDE QUALITÄT, FÜR INFORMATIONSGEHALT ODER RICHTIGKEIT, FÜR UNGESTÖRTE NUTZUNG, EIGENTUMSRECHTE UND DIE NICHTVERLETZUNG VON RECHTEN DRITTER, AB. ALTOVA ÜBERNIMMT KEINE GEWÄHRLEISTUNG FÜR DIE FEHLERFREIHEIT DER ACDE ODER ACBE SOFTWARE ODER, DASS DIESE OHNE UNTERBRECHUNGEN LAUFFÄHIG SIND. FALLS NACH ZWINGEND ANWENDBAREM RECHT IRGENDWELCHE GEWÄHRLEISTUNGEN IM BEZUG AUF ACBE ODER ACDE SOFTWARE VORGESEHEN SIND, WERDEN SOLCHEN GEWÄHRLEISTUNGEN FÜR EINEN ZEITRAUM VON 30 TAGEN AB DER INSTALLATION ODER DEM BEGINN DER VERWENDUNG DER SOFTWARE, JE NACHDEM WAS FRÜHER IST, EINGESCHRÄNKT. IN MANCHEN STAATEN ODER LÄNDERN IST DER AUSSCHLUSS VON IMPLIZITEN GEWÄHRLEISTUNGEN NICHT ERLAUBT, DAHER KÖNNTE DER OBEN ANGEFÜHRTE GEWÄHRLEISTUNGSAUSSCHLUSS SIE NICHT BETREFFEN. DIESE GEWÄHRLEISTUNG GIBT IHNEN SPEZIFISCHE RECHTE. SIE KÖNNTEN AUCH ANDERE RECHTE BESITZEN WELCHE VON STAAT ZU STAAT UND VON LAND ZU LAND UNTERSCHIEDLICH SIND. SIE STIMMEN ZU, ALLEIN FÜR DIE RICHTIGKEIT UND ANGEMESSSENHEIT DER ACDE ODER ACBE SOFTWARE FÜR DEN VON IHNEN IN AUSSICHT GENOMMENEN VERWENDUNGSZWECK VERANTWORTLICH ZU SEIN UND SIE WERDEN ALTOVA SCHAD- UND KLAGLOS HALTEN IN BEZUG AUF ALLE KLAGEN DRITTER SOWEIT DIESE AUF DER RICHTIGKEIT UND ANGEMESSENHEIT DER ACDE ODER ACBE SOFTWARE, DIE SIE BENUTZT HABEN, BASIEREN.

(c) Haftungseinschränkung. SOWEIT DIE ANWENDBAREN GESETZE DIES IM GRÖßTMÖGLICHEN AUSMAß ZULASSEN, HAFTEN ALTOVA ODER IHRE ZULIEFERER, IN KEINEM FALL FÜR KONKRETE, ZUFÄLLIG ENTSTANDENE, MITTELBARE, UNMITTELBARE ODER FOLGESCHÄDEN WELCHER ART AUCH IMMER (INSBESONDERE SCHÄDEN AUS ENTGANGENEM GEWINN, GESCHÄFTSUNTERBRECHUNGEN, VERLUST VON GESCHÄFTSINFORMATIONEN ODER ANDEREN FINANZIELLEN VERLUSTEN), DIE DURCH DIE NUTZUNG ODER DIE UNMÖGLICHKEIT DER NUTZUNG DER AUTHENTIC SOFTWARE ODER DIE NICHTERBRINGUNG VON SUPPORTLEISTUNGEN ENTSTANDEN SIND, ODER DIESES LIZENZVERTRAGES, SELBST WENN ALTOVA AUF DIE MÖGLICHKEIT EINES SOLCHEN SCHADENS AUFMERKSAM GEMACHT WURDE. DORT, WO DIE HAFTUNG RECHTLICH NICHT AUSGESCHLOSSEN ABER BESCHRÄNKT WERDEN KANN; IST ALTOVAS HAFTUNG; UND DIE IHRER ZULIEFERER AUF EINE SUMME VON 50 DOLLAR (USD $50) BESCHRÄNKT, DA IN MANCHEN STAATEN UND JURISDIKTIONEN DER AUSSCHLUSS ODER DIE BEGRENZUNG VON HAFTUNGEN NICHT MÖGLICH IST, IST DIE OBEN ANGEFÜHRTE HAFTUNGSBESCHRÄNKUNG AUF SIE VIELLEICHT NICHT ANWENDBAR, IN SOLCHEN STAATEN UND JURISDIKTIONEN WIRD DIE HAFTUNG VON ALTOVA UND IHREN ZULIEFERERN IM GRÖSSTMÖGLICHEN RECHTLICH ERLAUBTEN AUSMASS EINGESCHRÄNKT. DIE BISHER ANGEFÜHRTEN EINSCHRÄNKUNGEN DER HAFTUNG SIND ANZUWENDEN AUF DIE OBEN ANGEFÜHRTEN GEWÄHRLEISTUNGEN; GEWÄHRLEISTUNGSAUSSCHLÜSSE UND ALLE ANDEREN ASPEKTE DES LIZENZVERTRAGES:

(d) Support. Altova ist nicht verpflichtet technischen Support für ACDE oder ACBE Software, welche wie besehen zur Verfügung gestellt wird, zur Verfügung zu stellen. In dem Ausmaß in dem Altova aus eigener Entscheidung heraus Support für diese Produkte zur Verfügung stellt, wird dieser Support in der Art und Weise wie sie in Abschnitt 3(b) dieses Lizenzvertrages beschrieben wird und unter allen darin aufgeführten Voraussetzungen durchgeführt.

7. Authentic Software (AED, ABE, ACDE, und ACBE) Bestimmungen

 © 2010 Altova GmbH

Die in Abschnitt 7 festgehaltenen Bestimmungen sind auf AED, ABE, ACDE und ACBE Software Lizenzen anwendbar.

(a) Eigentum. Ihnen wird kein Eigentum an der Authentic Software übertragen. Das Eigentum an allen Kopien der Authentic Software und an allen Kopien der Authentic Software, welche Sie erstellen steht Altova zu, vorbehaltlich der Rechte auf Verwendung und Vertrieb, soweit anwendbar, die Ihnen in diesem Lizenzvertrag gewährt werden. Alle Rechte, welche Ihnen nicht durch diesen Lizenzvertrag ausdrücklich gewährt werden, sind Altova vorbehalten.

(b) Anerkennung der Rechte von Altova. Die Authentic Software und alle Kopien, deren Anfertigung Ihnen von Altova gestattet ist, sind das geistige Eigentum von Altova und dessen Lieferanten und stehen in deren Besitz. Struktur, Organisation und Code der Authentic Software stellen wertvolle Betriebsgeheimnisse und vertrauliche Informationen von Altova und dessen Lieferanten dar. Die Authentic Software ist durch gesetzliche Bestimmungen urheberrechtlich geschützt. Diese gesetzlichen Bestimmungen beinhalten (ohne darauf beschränkt zu sein) das Urheberrecht der Vereinigten Staaten, internationale Verträge und das in den Ländern, in denen die Authentic Software genutzt wird, geltende Recht. Sie anerkennen, dass sich Altova das Eigentum an allen Patenten, Urheberrechten, Branchengeheimnissen, Warenzeichen und sonstigen geistigen Eigentumsrechten, die hinsichtlich der Software bestehen, vorbehält. Das Eigentumsrecht von Altova erstreckt sich auch auf alle Bilder, Fotografien, Animationen, Videos, Audioaufzeichnungen, Musikstücke, Texte und „Applets“, die Teil der Authentic Software sind, und alle dazugehörigen Unterlagen in gedruckter Form. Sie dürfen keine Handlungen vornehmen, die sich nachteilig auf die geistigen Eigentumsrechte von Altova an der Authentic Software auswirken. Warenzeichen dürfen nur in Übereinstimmung mit den anerkannten Standards für die Verwendung von Warenzeichen (einschließlich der namentlichen Nennung der Warenzeicheninhaber) verwendet werden. Die Verwendung von Warenzeichen ist nur zur Kennzeichnung von Druckmaterialien, die mit der Authentic Software hergestellt wurden, gestattet. Es entstehen Ihnen daraus keinerlei Eigentumsrechte an dem betreffenden Warenzeichen. XMLSpy, Authentic, StyleVision, MapForce, Markup Your Mind, Axad, Nanonull und Altova sind (in mehreren Ländern eingetragene) Warenzeichen von Altova GmbH. Unicode und das Logo von Unicode sind Warenzeichen von Unicode, Inc. Windows, Windows 95, Windows 98, Windows NT, Windows 2000 und Windows XP sind Warenzeichen von Microsoft. W3C, CSS, DOM, MathML, RDF, XHTML, XML und XSL sind (in mehreren Ländern eingetragene) Warenzeichen des World Wide Web Consortium (W3C). Marken des W3C sind von den Dachinstitutionen des Konsortiums (MIT, INRIA und Keio) eingetragen und stehen in deren Eigentum. Sofern dies nicht ausdrücklich hierin festgelegt ist, entstehen Ihnen aus diesem Lizenzvertrag keinerlei geistigen Eigentumsrechte an der Authentic Software. Mitteilungen über geltend gemachte Verstöße gegen geistige Eigentumsrechte sind an den Copyright Agent von Altova zu richten (nähere Angaben dazu finden Sie auf der Website von Altova).

(c) Gemeinsame Einschränkungen.

(i) Sie sind nicht berechtigt, an der Software Reverse Engineering durchzuführen, die Software zu dekompilieren, zu disassemblieren, oder auf andere Weise zu versuchen, den Quellcode, zugrunde liegende Konzepte, Techniken für Benutzerschnittstellen oder Algorithmen der Authentic Software direkt oder indirekt zu entschlüsseln. Davon ausgenommen ist lediglich das ausdrücklich nach geltendem Recht der Europäischen Union gestattete Maß an Dekompilierung, das erforderlich ist, um die Funktionsfähigkeit der Authentic Software mit anderen Softwareprogrammen zu erzielen. Dies ist Ihnen jedoch erst dann gestattet, wenn Sie zuerst bei Altova die dafür notwendigen Informationen angefordert haben und Altova Ihnen diese nicht zur Verfügung gestellt hat. Altova ist berechtigt, für die Bereitstellung solcher Informationen angemessene Bedingungen aufzustellen und eine angemessene Entschädigung zu verlangen. Die von Altova zur Verfügung gestellten Informationen bzw. alle von Ihnen im Sinne der hier festgelegten Bestimmungen rechtmäßig erworbenen Informationen dürfen nur zu den hierin angeführten Zwecken verwendet und keinesfalls an Dritte weitergegeben oder zur Entwicklung von Software genutzt werden, die der Authentic Software in wesentlichen Aspekten ähnlich ist. Informationen im Sinne dieses Absatzes können von Benutzern aus der Europäischen Union bei der

Kundendienstabteilung von Altova angefordert werden. Sie sind nicht berechtigt, die Authentic Software (zur Gänze oder teilweise) zu verleihen, zu vermieten, per Leasing zur Verfügung zu stellen, weiterzulizenzieren, weiterzugeben oder auf sonstige Weise Dritten zu überlassen, es sei denn, dies ist durch diesen Lizenzvertrag ausdrücklich gestattet.

(ii) Sofern dies nicht ausdrücklich in diesem Vertrag gestattet ist, dürfen Sie die Authentic Software nicht kopieren. Alle Kopien, zu deren Anfertigung Sie Aufgrund dieses Lizenzvertrages berechtigt sind, müssen die im Original enthaltenen urheber- und patentrechtlichen Angaben sowie alle anderen Angaben hinsichtlich geistiger Eigentumsrechte unverändert enthalten. Die Authentic Software darf nicht verändert, adaptiert oder übersetzt werden. Sie dürfen weder direkt noch indirekt ein Pfand- oder Sicherungsrecht an der Authentic Software bestellen oder die Bestellung eines solchen zulassen. Es ist Ihnen nicht gestattet, wissentlich eine Handlung vorzunehmen, durch die die Authentic Software öffentlich zugänglich wird, oder die Authentic Software in einer Computerumgebung zu verwenden, die nicht in diesem Lizenzvertrag angegeben ist. Sie haben die geltenden gesetzlichen Bestimmungen und die Anweisungen von Altova in Bezug auf die Benutzung der Authentic Software einzuhalten. Sie sind verpflichtet, Ihre Mitarbeiter und Vertreter, die Zugriff auf die Authentic Software haben, von den in diesem Lizenzvertrag enthaltenen Beschränkungen in Kenntnis zu setzen und diese auf sie zu überbinden. Es ist Ihnen nicht gestattet die Authentic Software zu verändern oder zu modifizieren oder einen neuen Installer für die Authentic Software zu erstellen.

(d) Authentic Software AKTIVIERUNG, UPDATES, LIZENZÜBERWACHUNG, UND Datengebrauch

(i) Altova verfügt über ein integriertes Lizenzüberwachungsmodul, das Ihnen hilft, unbeabsichtigte Verletzungen dieses Lizenzvertrages zu vermeiden. Dabei kann Altova zur Lizenzüberwachung von verschiedenen installierten Versionen der Authentic Software Ihr internes Netzwerk verwenden. **Die Authentic Software von Altova kann Ihr internes Netzwerk und Ihre Internetverbindung verwenden, um Angaben über Ihre Lizenz im Zuge der Installation, Registrierung, Benutzung oder Aktualisierung an einen Altova-Lizenzserver zu übertragen und diese zu verifizieren, um auf diese Weise eine nicht lizenzierte oder illegale Verwendung der Authentic Software zu verhindern und den Kundenservice von Altova weiter zu verbessern. Die Aktivierung erfolgt über einen Austausch von lizenzbezogenen Daten zwischen Ihrem Computer und dem Altova-Lizenzserver. Sie stimmen dieser Vorgangsweise von Altova zu und verpflichten sich, allen diesbezüglichen Vorgaben Folge zu leisten. Sie erklären sich damit einverstanden, dass die Verwendung von Freischaltcodes, welche nicht von Altova erstellt werden oder wurden und nicht rechtmäßig von Altova oder einem dazu berechtigten Wiederverkäufer erworben wurden, zum Zweck der Aktivierung oder Nutzung der Authentic Software die gewerblichen Schutzrechte von Altova sowie die Bestimmungen dieses Lizenzvertrages verletzt. Sie erklären sich weiters damit einverstanden, dass Versuche mit dem Zweck der Umgehung oder Deaktivierung der urheberrechtlichen Schutzmaßnahmen oder Lizenzmanagementsystemen von Altova die gewerblichen Schutzrechte von Altova sowie die Bestimmungen dieses Lizenzvertrages verletzen. Altova behält sich ausdrücklich das Recht vor, alle rechtlich verfügbaren sowie angemessenen Mittel zu ergreifen, um derartige Praktiken zu verhindern und entgangenen Gewinn, Schäden und Kosten zurückerstattet zu bekommen.**

(ii) Altova stellt Ihnen ein neues LiveUpdate Benachrichtigungsservice zur Verfügung, welches kostenlos ist. Altova kann Ihr internes Netzwerk und Ihre Internetverbindung verwenden, um Angaben über Ihre Lizenz an einen LiveUpdate-Server von Altova zu übertragen, um Ihre Lizenz in gewissen Zeitabständen zu verifizieren und festzustellen, ob ein Update für Sie verfügbar ist. Der gesamte Wortlaut der Datenschutzbestimmungen von Altova kann unter http://www.altova.com/privacy eingesehen werden. Die Datenschutzbestimmungen sind durch Bezugnahme Teil dieses Lizenzvertrags. Durch Ihre Zustimmung zu den Bestimmungen dieses Lizenzvertrags bzw. durch Benutzung der Software erklären Sie sich damit einverstanden, dass Altova für die in diesem Lizenzvertrag und/oder in den Datenschutzbestimmungen (in der jeweils geltenden Fassung) genannten Zwecke Daten erhebt, verarbeitet und weitergibt. Benutzer in Europa haben Kenntnis davon und erklären sich damit einverstanden, dass Informationen zu ihrer Person in den Vereinigten Staaten zu den in diesem Vertrag

angeführten Zwecken verwendet werden. Altova behält sich das Recht vor, diese Bestimmung des Lizenzvertrags und/oder der Datenschutzbestimmungen jederzeit zu ändern. Wir legen Ihnen nahe, die auf der Website von Altova veröffentlichten Datenschutzbestimmungen von Zeit zu Zeit erneut durchzulesen.

(e) Keine weiteren Garantien, Haftungsausschluss. ES WIRD KEINE GARANTIE ODER GEWÄHRLEISTUNG FÜR DIE FEHLERFREIHEIT DER AUTHENTIC SOFTWARE ABGEGEBEN ODER GEWÄHRT UND VON ALTOVA KEINERLEI HAFTUNG DIESBEZÜGLICH ÜBERNOMMEN. UNABHÄNGIG VON IRGENDWELCHEN SUPPORTLEISTUNGEN NACH IRGENDEINEM TECHNISCHEN STANDARD IST DIE AUTHENTIC SOFTWARE KEINESFALLS FÜR DIE VERWENDUNG IN ODER IM ZUSAMMENHANG MIT KERNKRAFTANLAGEN, FLUGGERÄTENAVIGATION, KOMMUNIKATIONSSYSTEMEN ODER LUFTVERKEHRSKONTROLLEINRICHTUNGEN, MEDIZINISCHEN GERÄTEN ODER LEBENSERHALTUNGSSYSTEMEN, MEDIZINISCHEN ODER DIE GESUNDHEITSVORSORGE BETREFFENDEN ANWENDUNGEN, ODER SONSTIGEN ANWENDUNGEN KONZIPIERT, BEI DENEN FEHLER DER AUTHENTIC SOFTWARE ODER FEHLER BEI DER DATENVERARBEITUNG ZU TODESFÄLLEN, PERSONENSCHÄDEN ODER SCHWEREN SACH- ODER UMWELTSCHÄDEN FÜHREN KÖNNTEN. SIE ERKLÄREN SICH DAMIT EINVERSTANDEN, DASS SIE DIE ALLEINIGE VERANTWORTUNG FÜR DIE EIGNUNG UND ZWECKMÄSSIGKEIT DER AUTHENTIC SOFTWARE UND ALLE MIT DER SOFTWARE ERSTELLTEN ODER VERARBEITETEN DATEN FÜR DEN BEABSICHTIGTEN VERWENDUNGSZWECK TRAGEN UND SIE WERDEN ALTOVA UND DEREN GESCHÄFTSFÜHRER SOWIE ANGESTELLTE HINSICHTLICH ALLER ANSPRÜCHE, FORDERUNGEN UND KLAGEN DRITTER, DIE AUF DIE EIGNUNG UND ZWECKMÄSSIGKEIT DER VON IHNEN VERWENDETEN AUTHENTIC SOFTWARE ODER DIE VON DER AUTHENTIC SOFTWARE ERSTELLTEN DATEN GEGRÜNDET SIND, SCHAD- UND KLAGLOS HALTEN.

(f) EINGESCHRÄNKTE RECHTE UND EXPORTBESCHRÄNKUNGEN. Die Entwicklung der Authentic Software wurde ausschließlich privat finanziert. Bei der Software handelt es sich um kommerzielle Computersoftware, die mit eingeschränkten Rechten ausgestattet ist. Die Verwendung, Vervielfältigung oder Weitergabe der Software durch die US-Regierung, eine Behörde oder einen Kooperationspartner der US-Regierung unterliegt den Beschränkungen im Rahmen dieses Vertrags sowie den Beschränkungen nach FAR 12.211 und 12.212 (48 C.F.R. §12.211 und 12.212) bzw. DFARS 227.7202 (48 C.F.R. §227-7202). Dabei wird eine Lizenz für kommerzielle Computersoftware und kommerzielle Computerdokumentation an die US-Regierung als Endnutzer ausschließlich in Form einer Lizenz für kommerziell genutzte Güter erteilt, weshalb es sich bei den damit verbundenen Rechten um dieselben Rechte handelt, die allen anderen Endnutzern im Rahmen dieses Lizenzvertrages gewährt werden. Beim Hersteller handelt es sich um Altova GmbH, Rudolfsplatz, 13a/9, A-1010 Wien, Österreich/EU. Die Authentic Software und/oder Dokumentation darf ausschließlich unter Einhaltung aller anwendbaren Exportbestimmungen der Vereinigten Staaten von Amerika sowie des Landes, in dem Sie die Authentic Software erhalten haben, verwendet, exportiert oder wiedereingeführt werden. Insbesondere darf die Software und/oder Dokumentation nicht (i) in ein Land exportiert oder wiedereingeführt werden, über das die Vereinigten Staaten ein Embargo verhängt haben, oder einem Staatsangehörigen oder Bewohner eines solchen Landes überlassen werden; oder (ii) einer Person überlassen werden, die auf der Liste der Specially Designated Nationals des U.S. Treasury Department oder dem Table of Denial Orders des U.S. Department of Commerce verzeichnet sind. Indem Sie die Software benutzen, erklären Sie, dass Sie weder in einem dieser Länder ansässig sind noch seiner Kontrolle unterliegen, noch ein Staatsangehöriger oder Bewohner eines dieser Länder sind, noch auf einer der vorstehend erwähnten Listen genannt werden.

(g) Beendigung. Dieser Lizenzvertrag wird folgendermaßen beendet, ohne dass damit andere Rechte oder Rechtsbehelfe von Altova ausgeschlossen werden: (i) indem Sie Altova eine schriftliche Benachrichtigung von der Vertragsbeendigung übermitteln, oder (ii) durch Altova, in Ausübung dessen Rechtes, in Form einer schriftlichen Benachrichtigung, oder (iii) durch Altova in Form einer diesbezüglichen schriftlichen Benachrichtigung, wenn Sie gegen diesen Lizenzvertrag verstoßen. Dieser

Lizenzvertrag endet automatisch mit dem Ablauf des ABE Lizenz-Zeitraums. Bei Beendigung des Lizenzvertrags dürfen Sie die gesamte Authentic Software, die unter diesem Lizenzvertrag lizenziert wurde, nicht mehr verwenden, müssen alle Kopien, die in Ihrem Besitz oder Einflussbereich stehen, vernichten, und müssen in zumutbarem Rahmen alle von Altova geforderten Maßnahmen ergreifen, um sicherzustellen, dass keine Kopien der Authentic Software in Ihrem Besitz oder Einflussbereich verbleiben. Die in den Abschnitten 1(e), 2(c)-(d), 3(c)-(d), 4(c)-(d), 5(b)-(c), 6(b)-(c) und 7, niedergelegten Bestimmungen, soweit anwendbar, bleiben auch nach Beendigung dieses Lizenzvertrags weiterhin aufrecht.

(h) SOFTWARE DRITTER. Die dieser Lizenz unterliegende Authentic Software kann Software Dritter enthalten, für die ergänzende Vermerke und/oder Nutzungsbedingungen gelten. Diese Vermerke und ergänzenden Nutzungsbedingungen für die Software Dritter können über unsere Website unter http://www.altova.com/legal_3rdparty.html eingesehen werden und sind durch Bezugnahme Teil dieses Lizenzvertrages. Indem Sie Ihre Zustimmung zu den Bedingungen dieses Lizenzvertrages erteilen, stimmen Sie auch automatisch allen ergänzenden Bestimmungen, die möglicherweise darin enthalten sind, zu.

(i) ALLGEMEINE BESTIMMUNGEN. Dieser Lizenzvertrag enthält die gesamte Vereinbarung zwischen den Parteien in Bezug auf den Vertragsgegenstand und tritt an die Stelle aller diesbezüglichen früheren mündlichen oder schriftlichen Vereinbarungen zwischen den Parteien. Benachrichtigungen oder sonstige Mitteilungen im Rahmen dieses Lizenzvertrages müssen schriftlich erfolgen und gelten als ordnungsgemäß übermittelt, wenn sie per Einschreiben mit Rückschein oder per Kurierdienst mit Zustellung am nächsten Tag an die auf der Website von Altova angegebene Adresse (wenn Altova der Empfänger ist) bzw. an die in den Aufzeichnungen von Altova für Sie eingetragene Adresse (wenn Sie der Empfänger sind) oder aber an eine andere, von den Vertragspartnern festgelegte und auf die vorstehend beschriebene Weise bekannt gegebene Adresse geschickt werden. Dieser Lizenzvertrag ist für die Vertragspartner verbindlich und geht auf ihre Erben, persönlichen und rechtlichen Vertreter, verbundenen Unternehmen, Rechtsnachfolger und zulässigen Abtretungsempfänger über. Die Nichtdurchsetzung oder Nichtausübung von Rechten oder Rechtsmitteln unter diesem Lizenzvertrag durch die Vertragspartner stellt keinen Verzicht auf ein solches Recht oder Rechtsmittel dar und beeinträchtigt in keiner Weise das Recht des betreffenden Vertragspartners, ein solches Recht oder Rechtsmittel sowie alle anderen Rechte oder Rechtsmittel aus diesem Lizenzvertrag später durchzusetzen bzw. auszuüben. Eine Änderung dieses Lizenzvertrages ist nur Form einer schriftlich niedergelegten Vereinbarung möglich, die von beiden Vertragspartnern unterzeichnet wird. Wenn eine Zuwiderhandlung gegen die Bestimmungen dieses Lizenzvertrages durch einen der Vertragspartner erfolgt ist oder droht, so kann der jeweils andere Vertragspartner alle ihm zustehenden Rechte und Rechtsmittel geltend machen. Jeder Vertragspartner ist ordnungsgemäß befugt und ermächtigt, in diesen Lizenzvertrag einzutreten und die daraus erwachsenden Verpflichtungen zu erfüllen. Sollte eine Bestimmung dieses Lizenzvertrages für unwirksam, rechtswidrig oder undurchführbar erklärt werden, so wird dadurch die Wirksamkeit der übrigen Bestimmungen nicht berührt, und dieser Lizenzvertrag bleibt im größtmöglichen rechtlich zulässigen Umfang wirksam und gültig. Die Vertragspartner haben die vorstehenden Vertragsbedingungen zur Kenntnis genommen und erklären sich ausdrücklich damit einverstanden.

(i) Wenn Sie sich in der Europäischen Union befinden und die Authentic Software in der Europäischen Union und nicht in den Vereinigten Staaten verwenden, unterliegt dieser Lizenzvertrag dem Recht der Republik Österreich (unter Ausschluss von dessen Verweisungsnormen und der UN-Kaufrechtskonvention). Sie erklären sich ausdrücklich damit einverstanden, dass alle Streitigkeiten oder Konflikte mit Altova, die in Zusammenhang mit Ihrer Verwendung der Authentic Software stehen, in die alleinige Zuständigkeit des Handelsgerichts Wien fallen. Sie erklären sich weiters ausdrücklich damit einverstanden, dass Sie bezüglich solcher Streitigkeiten oder Konflikte der persönlichen Zuständigkeit des Handelsgerichts Wien unterstellt sind.

(ii) Wenn Sie sich in den Vereinigten Staaten befinden und die Authentic Software in den Vereinigten Staaten verwenden, unterliegt dieser Lizenzvertrag dem Recht des Commonwealth of Massachusetts, USA (unter Ausschluss von dessen Verweisungsnormen und der UN-Kaufrechtskonvention). Sie erklären sich ausdrücklich damit einverstanden, dass alle Streitigkeiten

 © 2010 Altova GmbH

oder Konflikte mit Altova, die in Zusammenhang mit Ihrer Verwendung der Authentic Software stehen, in die alleinige Zuständigkeit der einzel- und bundesstaatlichen Gerichte im Bundesstaat Massachusetts fallen. Sie erklären sich weiters ausdrücklich damit einverstanden, dass Sie bezüglich solcher Streitigkeiten oder Konflikte der persönlichen Zuständigkeit der einzel- und bundesstaatlichen Gerichte im Bundesstaat Massachusetts unterstellt sind.

(iii) Wenn Sie sich nicht in der Europäischen Union oder den Vereinigten Staaten befinden und die Authentic Software nicht in den Vereinigten Staaten verwenden, unterliegt dieser Lizenzvertrag dem Recht der Republik Österreich (unter Ausschluss von dessen Verweisungsnormen und der UN-Kaufrechtskonvention). Sie erklären sich ausdrücklich damit einverstanden, dass alle Streitigkeiten oder Konflikte mit Altova, die in Zusammenhang mit Ihrer Verwendung der Authentic Software stehen, in die alleinige Zuständigkeit des Handelsgerichts Wien fallen. Sie erklären sich weiters ausdrücklich damit einverstanden, dass Sie bezüglich solcher Streitigkeiten oder Konflikte der persönlichen Zuständigkeit des Handelsgerichts Wien unterstellt sind. Die Anwendung von Verweisungsnormen einer Rechtsordnung sowie des UN-Abkommens zum internationalen Warenkauf (CISG) auf diesen Lizenzvertrag wird ausdrücklich ausgeschlossen.

Letzte Aktualisierung: 2009-10-07

Index

A

AuthenicRange-Objekt, 65
Authentic, 145
attachCallBack, 86
ControlInitialized, 93
CreateChild, 94
CurrentSelection, 95
DesignDataLoadObject, 96
EditClear, 97
EditCopy, 98
EditCut, 99
EditPaste, 100
EditRedo, 101
EditSelectAll, 102
EditUndo, 103
event, 109
FindDialog, 110
FindNext, 111
GetAllAttributes, 112
GetAllowedElements, 114
GetFileVersion, 116
GetNextVisible, 117
GetPreviousVisible, 118
IsEditClearEnabled, 119
IsEditCopyEnabled, 120
IsEditCutEnabled, 121
IsEditPasteEnabled, 122
IsEditRedoEnabled, 123
IsEditUndoEnabled, 124
IsFindNextEnabled, 125
IsRowAppendEnabled, 126
IsRowDeleteEnabled, 127
IsRowDuplicateEnabled, 128
IsRowInsertEnabled, 129
IsRowMoveDownEnabled, 130
IsRowMoveUpEnabled, 131
IsTextStateApplied, 132
IsTextStateEnabled, 133
LoadXML, 134
MarkUpView, 135
Print, 137
PrintPreview, 138
ReplaceDialog, 141
Reset, 142
RowAppend, 143
RowDelete, 144
RowInsert, 146
RowMoveDown, 147
RowMoveUp, 148
Save, 149
SavePOST, 151
SaveXML, 152
SchemaLoadObject, 153
SelectionChanged, 154
SelectionMoveTabOrder, 155
SelectionSet, 156
StartEditing, 158
TextState anwenden, 85
ValidateDocument, 166
validationBadData, 167
validationMessage, 168
XMLDataLoadObject, 169
XMLDataSaveUrl, 170
XMLRoot, 171
Authentic Browser, 82
Class IDs, 16, 28
Datei-Download auf den Server, 14
Datei-Download auf Server, 15, 16
Event-Behandlung, 32
MIME Types, 16, 40
Netzwerk einrichten, 10
Plug-in Dile, 14
Plug-in-Datei, 15, 16
Projektimplementierung, 11
Subroutinen, 32
Übersicht, 10
und datenbankbasierte SPSs, 22
Version, 28, 40
Versionen, 16
Vorteile, 9
Authentic RowDuplicate, 145
AuthenticDataTransfer,
dropEffect, 182
getData, 183
ownDrag, 184
type, 185
AuthenticEvent,
altKey, 187
altLeft, 188

© 2010 Altova GmbH

AuthenticEvent,
button, 189
cancelBubble, 190
clientX, 191
clientY, 192
ctrlKey, 193
ctrlLeft, 194
dataTransfer, 195
fromElement, 196
keyCode, 197
propertyName, 198
repeat, 199
returnValue, 200
shiftKey, 201
shiftLeft, 202
srcElement, 203
type, 204
Authentic-Objekt, 65
AuthenticRange, 208
AppendRow, 210
Application, 211
CanPerformAction, 212
CanPerformActionWith, 213
Close, 214
CollapsToBegin, 215
CollapsToEnd, 216
Copy, 217
Cut, 218
Delete, 219
DeleteRow, 220
DuplicateRow, 221
ExpandTo, 222
FirstTextPosition, 223
FirstXMLData, 224
FirstXMLDataOffset, 225
GetElementAttributeNames, 227
GetElementAttributeValue, 228
GetElementHierarchy, 229
GetEntityNames, 230
Goto, 231
GotoNext, 232
GotoNextCursorPosition, 233
GotoPrevious, 234
GotoPreviousCursorPosition, 235
HasElementAttribute, 236
InsertEntity, 237
InsertRow, 238
IsEmpty, 239
IsEqual, 240
IsInDynamicTable, 241
LastTextPosition, 242
LastXMLData, 243
LastXMLDataOffset, 244
MoveBegin, 246
MoveEnd, 247
MoveRowDown, 249
MoveRowUp, 248
Parent, 250
Paste, 251
PerformAction, 252
Select, 253
SelectNext, 254
SelectPrevious, 255
SetElementAttributeValue, 256
SetFromRange, 257
Text, 258
AuthenticView, 88, 281, 292
Application, 282
DocumentBegin, 283
DocumentEnd, 284
Goto, 285
MarkupVisibility, 286
Parent, 287
Print, 288
Redo, 289
Selection, 290
Undo, 291
WholeDocument, 293
AuthenticView-Objekt, 65

B

Bearbeitungsoperationen, 66
Benutzerreferenz, 58
Browser,
auf dem Client ermitteln, 51
Browser-Dienst für Server, 17

C

CAB-Datei, 14, 15, 16
Class IDs, 16, 28

© 2010 Altova GmbH

CODEBASE Attribut, 28
Connection Point-Ereignisse, 60
ControlInitialized, 60, 93
Copyright-Informationen, 368, 370

D

Datenbankbasiertes SPS,
Voraussetzungen für Authentic Browser, 22
DB-basiertes SPS, 38
Dokumentinhalt,
aufrufen und ändern, 65
DOM,
und XMLData, 78
Dynamische Tabellen, 68

E

Eingabehilfen, 71
EMBED-Element,
in HTML-Seite für Firefox, 40
Endbenutzer-Lizenzvereinbarung, 368, 371
Ereignishandler, 60, 63
Ereignisse,
Referenz, 64
Ersetzen, 67
Evaluierungszeitraum,
für Altova-Software-Produkte, 369
von Altova Software-Produkten, 368
Event Handling, 32

H

HTML-Seite,
Übersicht, 24
HTML-Seite für Firefox,
Beispiel für das Sortieren einer Tabelle, 47
einfaches Beispiel, 45
EMBED-Element, 40
Hinzufügen von Event Listener-Funktionen, 44
Übersicht, 39
HTML-Seite für IE,
Beispiel zum Sortieren einer Tabelle, 35
DB-basiertes Beispiel, 38
einfaches Beispiel, 33
OBJECT-Element, 28
SCRIPT-Element, 32
Übersicht, 27
HTML-Seite für IE oder Firefox,
Beispieldatei, 53
Ermitteln der Browser-Version des Client, 51
Übersicht, 50

I

Inhalt,
aufrufen und ändern, 65
Internet Information Service für Server, 17

L

Lizenz, 371
Informationen, 368
Lizenzierung für die Enterprise Edition, 25

M

MIME Types, 16, 40

N

Netzwerk einrichten, 10

O

OBJECT-Element,
in HTML-Seite für IE, 28

P

Pakete, 72

© 2010 Altova GmbH

R

Rechtliches, 368
Rechtschreibprüfung, 72
Referenz,
Ereignisse, 64

S

Schritte bei der Projekterstellung, 11
SCRIPT-Element,
in HTML-Seite für IE, 32
selectionchanged, 60
Server einrichten, 14
Software-Produktlizenz, 371
Subroutinen, 32
Suchen, 67
Suchen und ersetzen, 67
Symbolleisten-Schaltflächen,
Verhalten ändern, 63
Systemvoraussetzungen, 10

T

Tastaturkürzel, 69
Textstatus-Schaltflächen, 70

V

Vertrieb,
von Altova Software-Produkten, 368
von Altova-Software-Produkten, 369

X

XMLData, 74
GetChild, 341
GetChildKind, 342
HasChildrenKind, 347
IsSameNode, 349
und DOM, 78
XMLSpyLib,
AuthenticDataTransfer, 181
AuthenticEvent, 186
XMLSpyXMLData, 335
XMLSPYPLUGINLib,
Authentic, 82
XMLSpyXMLLoadSave, 205
XMLSpyXMLData,
AppendChild, 336
EraseAllChildren, 339
EraseCurrentChild, 340
GetCurrentChild, 343
GetFirstChild, 344
GetNextChild, 345
HasChildren, 346
InsertChild, 348
Kind, 350
MayHaveChildren, 351
Name, 352
Parent, 353
TextValue, 354
XMLSpyXMLLoadSave,
String, 206
URL, 207
XPI-Datei, 14, 15, 16

Z

Zeilenoperationen, 68

© 2010 Altova GmbH